AD622330

AFCRL-65-555

OPTIMIZATION OF THE FIALKOW-GERST MULTIPORT RC TRANSFER FUNCTION SYNTHESIS

BY D. HAZONY AND D. HILBERMAN

CASE INSTITUTE OF TECHNOLOGY
UNIVERSITY CIRCLE
CLEVELAND, OHIO 44106

CONTRACT NO. AF19(628)1699
PROJECT NO. 5635
TASK NO. 563501

Scientific Report Number 10

1965

PREPARED FOR

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES

OFFICE OF AEROSPACE RESEARCH

UNITED STATES AIR FORCE

BEDFORD, MASS.

AFCRL-65-555

OPTIMIZATION OF THE FIALKOW-GERST MULTIPORT RC TRANSFER FUNCTION SYNTHESIS

by

D. Hazony

and

D. Hilberman

CASE INSTITUTE OF TECHNOLOGY

University Circle

Cleveland 6, Ohio

Contract No. AF19(628)1699

Project No. 5635

Task No. 563501

Scientific Report No. 10

1965

Prepared for

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS

Requests for additional copies by agencies of the Department of Defense, their contractors, or other government agencies should be directed to:

Defense Documentation Center (DDC)
Cameron Station
Alexandria, Virginia 22314

Department of Defense contractors must be established by DDC services or have their "need to know" certified by the cognizant military agency of their project or contract.

All other persons and organizations should apply to the:

Clearinghouse for Federal Scientific
and Technical Information (CFSTI)
Sills Building
5285 Port Royal Road
Springfield, Virginia 22151

OPTIMIZATION OF THE FIALKOW-GERST MULTIPORT RC TRANSFER FUNCTION SYNTHESIS*

D. Hazony and D. Hilberman

ABSTRACT

A method is presented which reduces the number of components needed in a Fialkow-Gerst multiport RC transfer function synthesis. A relationship is determined between the number of non-zero numerator coefficients in the transfer function vector and the number of components used in the synthesis of that vector.

*Work also supported by the American Society for Engineering Education and the Leeds and Northrup Company.

TABLE OF CONTENTS

INTRODUCTION

Although the Fialkow-Gerst synthesis technique has been known for some time[1-4], and although it has been extended to multiport networks by Zeren and others[5,9], the large number of components it uses is still a major problem. Kodali[8] has worked on reducing this number and an extension of his work will be presented which reduces the number of components in three ways. First, it eliminates components from the termination of the network. Second, it permits the calculation of an arbitrary constant, λ, arising in the Fialkow-Gerst synthesis. Finally, the method yields transfer functions which are frequently amenable to special-case synthesis techniques.

The importance of RC transfer function synthesis has been increased by the recent work of Hazony and Joseph[7], which permits the synthesis of any RLC transfer function vector with one unity gain amplifier and an RC network.

A computer program is provided in the Appendix which incorporates this variation of the Fialkow-Gerst synthesis.

CHAPTER I

THE FIALKOW-GERST TRANSFER FUNCTION SYNTHESIS

1. Synthesis of a Transfer Function Matrix

The Fialkow-Gerst synthesis of a grounded multiport RC transfer function network is well known[1-6] and will be outlined in this chapter only to introduce notation and provide a reference for later discussion.

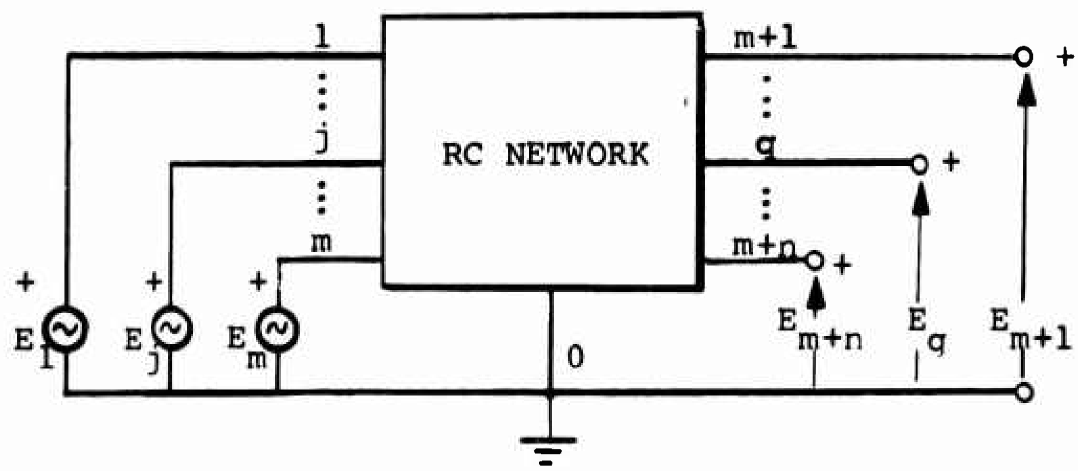


Fig. 1.1 A multiport RC network.

We can describe the network of Figure 1.1 by the equation

$$[E_{out}] = [T] \cdot [E_{in}] \ ,$$

where the elements, or entries, of the matrix T are given by

$$(1.1) \qquad t_{qj}(s) = \frac{a_{0qj}s^r + a_{1qj}s^{r-1} + \ldots + a_{rqj}}{b_{0q}s^r + b_{1q}s^{r-1} + \ldots + b_{rq}}$$

$$= \frac{A_{qj}(s)}{B_q(s)} = - \frac{Y_{qj}}{Y_{qq}}$$

for $q = m+1, \ldots, m+n$, and for $j = 1, \ldots, m$.

If a row of [T] is considered as a vector $\underline{t}$ then with appropriate surplus factors, $\underline{t}$ can be made to have only positive coefficients[1,6]. Each entry t_{qj} must then satisfy:

i) The poles are distinct and lie on the negative real axis;

$$\left.\begin{array}{lll} (1.2a) & ii) & 0 \leq \sum_j a_{iqj} \leq b_{iq} \\ (1.2b) & iii) & 0 \leq a_{iqj} \end{array}\right\} \begin{array}{l} \text{for } q = m+1, \ldots, m+n \\ i = 0, \ldots, r, \\ j = 1, \ldots, m. \end{array}$$

Such t_{qj} are termed RC R-functions and a matrix with all RC R-function entries is called an RC t-matrix.

For convenience we will assume throughout this paper that

$$b_o b_r \neq 0.$$

Following Fialkow, et al.[5], we synthesize only one row of an RC t-matrix at a time. If that q^{th} row is given by the vector

$$\underline{t}_q = (t_{q1}, t_{q2}, \ldots, t_{qm}) = \left[\frac{A_{q1}}{B_q}, \ldots, \frac{A_{qm}}{B_q}\right]$$

of degree* r with m vector entries, then the first step in the synthesis is to choose a polynomial D_q (of degree (r-1) with negative real simple zeros) such that

$$Y_{qq} = \frac{B_q}{D_q} = \frac{b_{0q}s^r + b_{1q}s^{r-1} + \ldots + b_{rq}}{d_{0q}s^{r-1} + \ldots + d_{r-1,q}}$$

and

$$-Y_{qj} = \frac{A_{qj}}{D_q}, \qquad \text{for all } j,$$

are RC admittances.

The second step is to split the short circuit driving-point admittance, Y_{qq}, into two such functions, Y'_{qq} and Y''_{qq}. Thus for the first cycle

---

*In this paper we are only concerned with RC transfer function vectors whose entries have a common denominator. Thus the term "degree" is used only in reference to that denominator and not the transfer function matrix as a whole.[6]

$$Y_{qq} = \frac{1}{\frac{1}{C_1 s} + \frac{1}{Y_{qq}^{(1)}}} + \frac{1}{R_2 + \frac{1}{Y_{qq}^{(2)}}} .$$

Obviously any method which accomplishes this split is valid. A method by Hazony[6,8], which will be used later, utilizes a split factor λ so that

$$(1.3) \qquad Y_{qq} = \left[\frac{b_{0q}s}{d_{0q}} + \lambda\left(\frac{g_{0q}s^{r-1} + \ldots + g_{r-2,q}s}{d_{0q}s^{r-1} + \ldots + d_{r-1,q}}\right)\right]$$

$$+\left[\frac{b_{rq}}{d_{r-1,q}} + (1-\lambda)\left(\frac{g_{0q}s^{r-1} + \ldots + g_{r-2,q}s}{d_{0q}s^{r-1} + \ldots + d_{r-1,q}}\right)\right]$$

where $0 < \lambda < 1$ but otherwise λ is arbitrary and in general it will have a different value for each synthesis cycle.

Using equation (1.3) we can write the new admittance $Y_{qq}^{(1)}$ as

$$(1.4a) \qquad Y_{qq}^{(1)} = \frac{b_{0q}^{(1)}s^{r-1} + b_{1q}^{(1)}s^{r-2} + \ldots + b_{r-1,q}^{(1)}}{d_{oq}^{(1)}s^{r-2} + \ldots + d_{r-2,q}^{(1)}} ,$$

where

(1.4b) $$b_{0q}^{(1)} = b_{0q} \; ,$$

(1.4c) $$b_{iq}^{(1)} = b_{iq}\lambda + \frac{b_{0q}d_{iq}(1-\lambda)}{d_{0q}} - \frac{\lambda b_{rq}d_{i-1,q}}{d_{r-1,q}} \; ,$$

for i = 1, ... , r-1, and

(1.4d) $$d_{iq}^{(1)} = d_{iq} - \frac{b_{iq}^{(1)}d_{r-1,q}}{b_{r-1,q}^{(1)}} \; ,$$

for i = 0, ... , r-2.

Similarly we can write $Y_{qq}^{(2)}$ as

(1.5a) $$Y_{qq}^{(2)} = \frac{b_{1q}^{(2)}s^{r-1} + b_{2q}^{(2)}s^{r-2} + \ldots + b_{rq}^{(2)}}{d_{1q}^{(2)}s^{r-2} + \ldots + d_{r-1,q}^{(2)}} \; ,$$

where, for i = 1, ... , r-1,

(1.5b) $$b_{iq}^{(2)} = b_{iq}(1-\lambda) - \frac{b_{0q}d_{iq}(1-\lambda)}{d_{0q}} + \frac{\lambda b_{rq}d_{i-1,q}}{d_{r-1,q}} \; ,$$

(1.5c) $$b_{rq}^{(2)} = b_{rq} \; ,$$ and

(1.5d) $$d_{iq}^{(2)} = d_{iq} - \frac{b_{i+1,q}^{(2)}d_{0q}}{b_{1q}^{(2)}} \; .$$

The resistor and capacitor removed are given by

(1.6a) $$C_1 = b^{(1)}_{r-1} \;/\; d_{r-1,q} \quad \text{farads and}$$

(1.6b) $$R_2 = d_{oq} \;/\; b^{(2)}_{1q} \quad \text{ohms.}$$

These equations emphasize the adaptability of this synthesis method to a computer solution.

The final step in the synthesis cycle is the computation of the two reduced transfer function vectors such that each vector entry, as well as the sum of the entries, is an RC R-function. Hence, for the first cycle $\underline{t}$ becomes

$$\underline{t}_q^{(1)} = \left[\frac{A^{(1)}_{q1}}{B^{(1)}_q} , \ldots , \frac{A^{(1)}_{qm}}{B^{(1)}_q} \right]$$

and

$$\underline{t}_q^{(2)} = \left[\frac{A^{(2)}_{q1}}{B^{(2)}_q} , \ldots , \frac{A^{(2)}_{qm}}{B^{(2)}_q} \right] ,$$

where the numerator coefficients satisfy

(1.7a) $$a_{0qj} = a^{(1)}_{0qj} , \quad a_{rqj} = a^{(2)}_{rqj} ,$$

$$a_{iqj} = a^{(1)}_{iqj} + a^{(2)}_{iqj} , \tag{1.7b}$$

for $i = 1, \ldots , r-1$, and all j,

and the denominator coefficients satisy

$$b_{0q} = b^{(1)}_{0q} , \quad b_{rq} = b^{(2)}_{rq} , \tag{1.7c}$$

$$b_{iq} = b^{(1)}_{iq} + b^{(2)}_{iq} , \text{ for } i = 1, \ldots , r-1. \tag{1.7d}$$

Equation (1.7) can be represented as

$$A_{qj} = s\ A^{(1)}_{qj} + A^{(2)}_{qj} , \quad \text{for all } j, \text{ and}$$

$$B_q = s\ B^{(1)}_q + B^{(2)}_q .$$

Obviously, to preserve the RC R-function characteristics, the new coefficients must satisfy equation (1.2). Hazony[5,6] has introduced a proportional method for calculating the reduced transfer function vector numerators which uses the following two equations:

$$\frac{a_{iqj}}{b_{iq}} = \frac{a'_{iqj}}{b'_{iq}} = \frac{a''_{iqj}}{b''_{iq}} , \tag{1.8a}$$

for $i = 1, \ldots , r-1$, and all j,

and, by equation (1.7a),

$$(1.8b) \qquad a_{0qj}^{(1)} = a_{0qj} \;, \quad a_{rqj}^{(2)} = a_{rqj} \;, \quad \text{for all } j.$$

Another method of calculating the transfer functions will be presented in Chapter 2.

After one cycle the vector network looks like Figure 1.2, where $\Gamma_q^{(1)}$ is a network with a transfer function vector $\underline{t}_q^{(1)}$ and a driving-point admittance $Y_{qq}^{(1)}$; $\Gamma_q^{(2)}$ is a similar network.

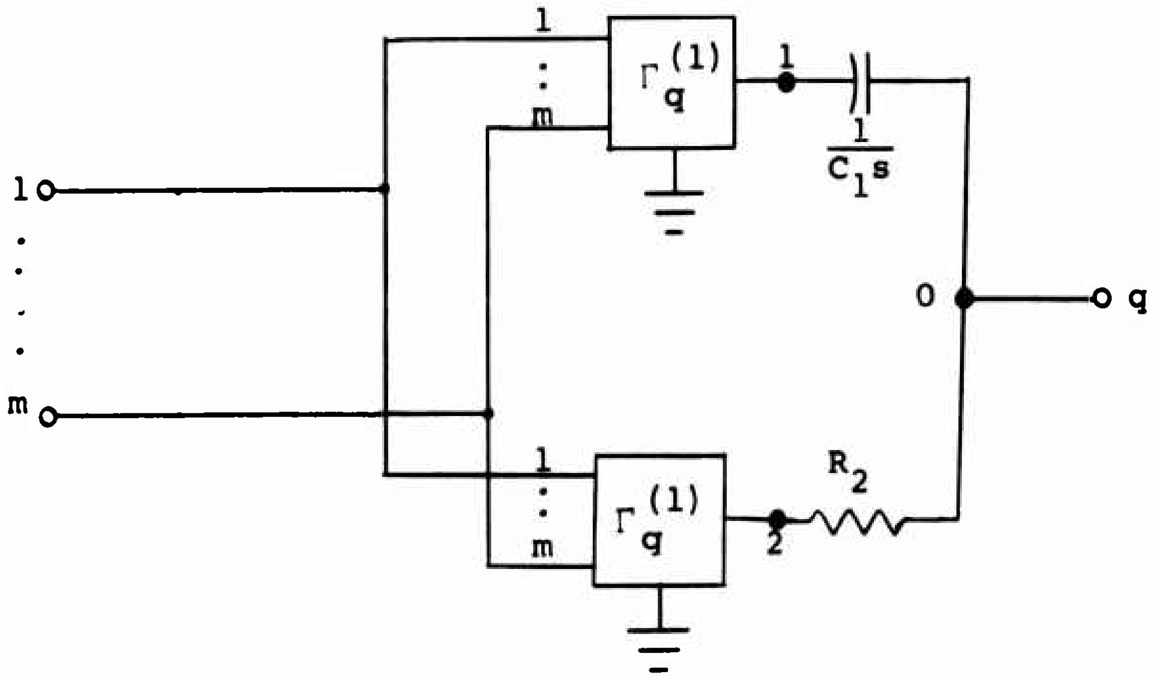


Fig. 1.2 The results of one synthesis cycle.

The above process is repeated until unity degree transfer functions are obtained, at which time the

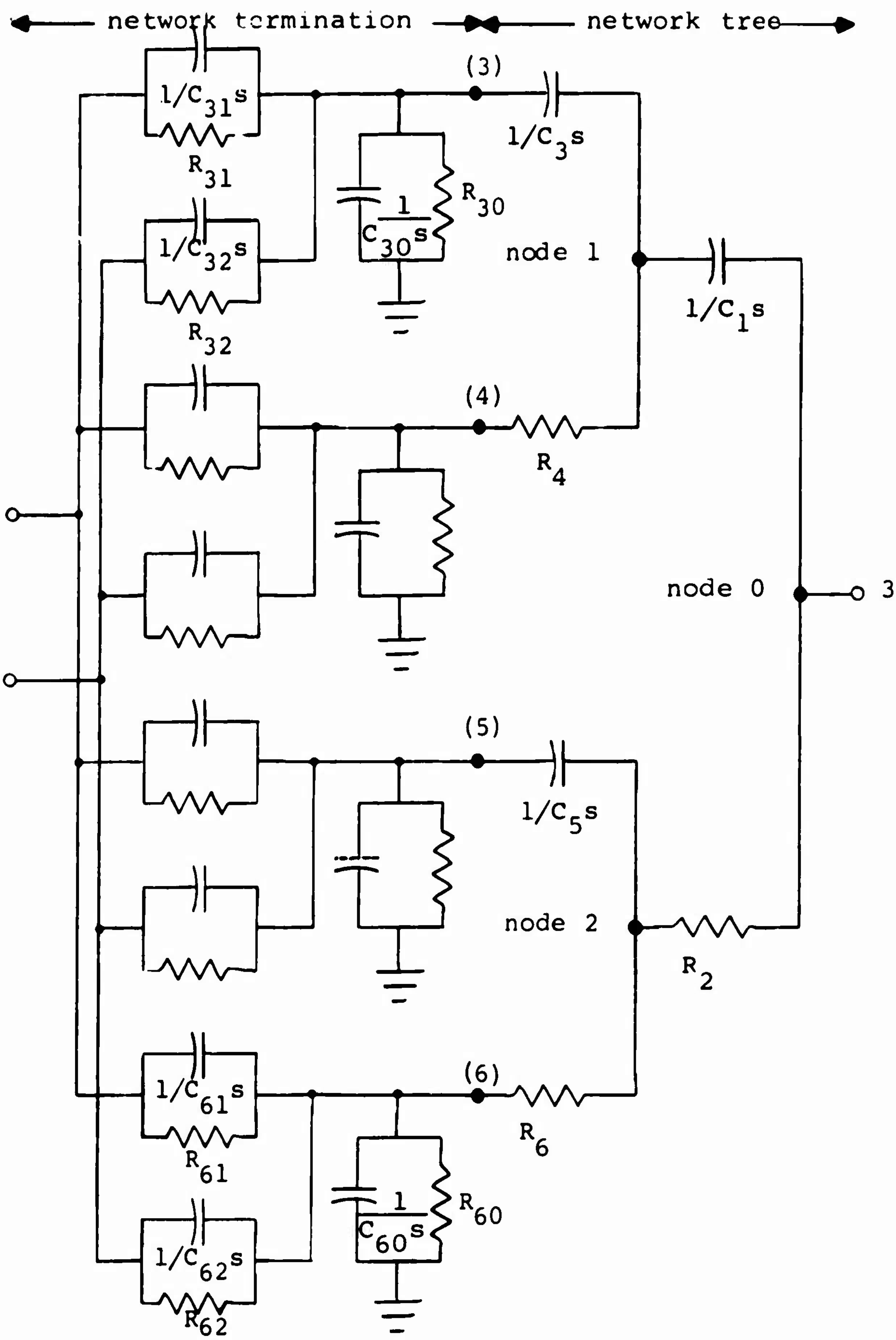


Fig. 1.3 A typical transfer function synthesis for degree three with two vector entries.

functions are synthesized directly and the network will appear as in Figure 1.3. In the subsequent discussion we will refer to the set of resistors and capacitors obtained from splitting the driving-point admittance as the <u>tree</u> of the network and the components obtained from the unity degree transfer functions as the network <u>termination</u>.

Once all of the row vectors have been synthesized, all of the j^{th} inputs are connected in parallel to yield one common j^{th} input to the whole network. As there is only one q^{th} output there is no interconnection between the output terminals. The interconnection takes the form of Figure 1.4.

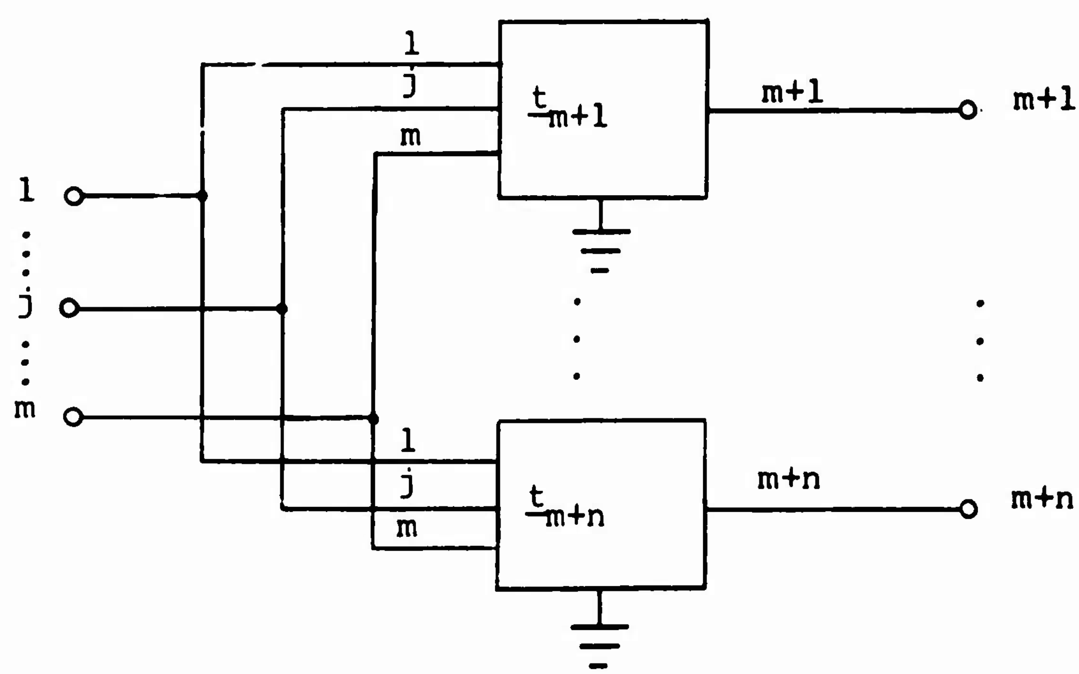


Fig. 1.4 Interconnection of synthesized vectors.

2. The Augumented Transfer Function Vector

When synthesizing the unity degree transfer function vectors the grounded components are values proportional to the differences between the denominator coefficients and the sums of the numerator coefficients. It is frequently convenient to make this "transfer function" to ground explicit: the complementary transfer function[1], t_{q0}, is given by $t_{q0} \equiv 1 - \sum_j t_{qj}$ and the augmented transfer function vector, $\underline{t}_q^+$, is the vector $\underline{t}_q$ with the entry t_{q0} added, i.e., $\underline{t}_q^+ \equiv (t_{q0}, t_{q1}, \ldots, t_{qm})$.

Since the augmented vector has the property that

$$\sum_{j=0}^{m} t_{qj} = 1 \ , \tag{1.9}$$

there are no components to "ground" when $\underline{t}_q^+$ is synthesized. Instead, there are components to the 0^{th} input port and that port has a voltage generator of zero volts connected to it.

Let us briefly examine the effect of interchanging one of the input terminals with the connection to ground. Using Figure 1.1, Figure 1.5a can be described by the equation

$$E_{q0} = t_{q1}E_{10} + \ldots + t_{qj}E_{j0} + \ldots + t_{qm}E_{m0} \ .$$

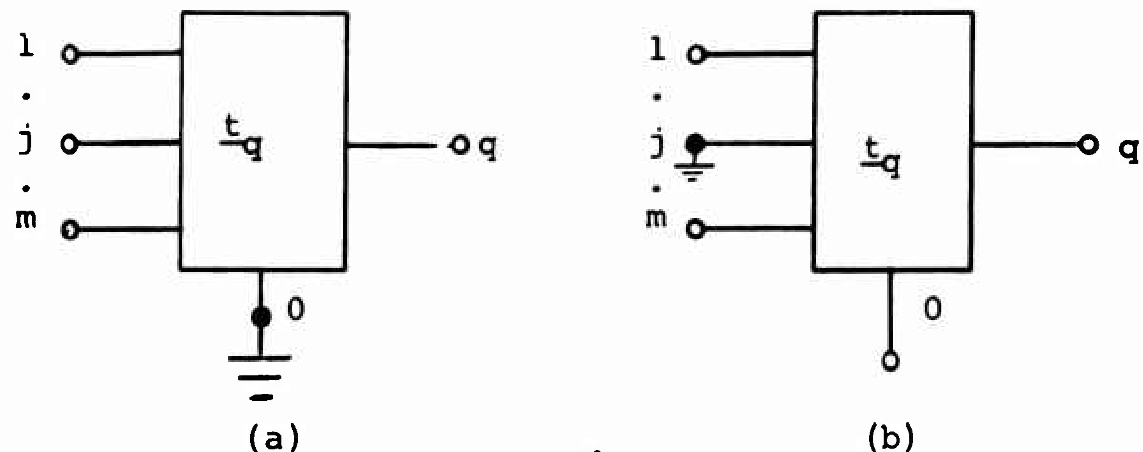


Fig. 1.5 Exchanging the j^{th} input with ground.

Subtracting E_{j0} from both sides and then adding and subtracting $t_{qj}E_{j0}$ on the right side yields (recall that $E_{ij} = E_{i0} - E_{j0}$)

$$E_{qj} = t_{q1}E_{1j} + t_{q2}E_{2j} + \dots + (t_{qj}-1-t_{qj})E_{j0}$$

$$+ \dots + t_{qm}E_{mj} + E_{j0}\sum_{i=1}^{m} t_{qi}$$

$$= t_{q1}E_{1j} + t_{q2}E_{2j} + \dots + (1-\sum_{i=1}^{m} t_{qi})E_{0j}$$

$$+ \dots + t_{qm}E_{mj}$$

$$= t_{q1}E_{1j} + t_{q2}E_{2j} + \dots + t_{q0}E_{0j}$$

$$+ \dots + t_{qm}E_{mj} \quad ,$$

which describes the transfer function vector for the new network of Figure 1.5b in which the j^{th} terminal is common.

It is apparent that an exchange of the j^{th} input terminal with ground amounts to interchanging t_{q0} with t_{qj} in $\underline{t}_q^+$, which is really no more than renaming the terminals. This will be used later to justify arbitrary rearrangements of the augmented transfer function vector entries.

When t_{qj} is zero, there is no connection to the j^{th} terminal and therefore the vector can be regarded as an (m-1) entry vector. This fact will be utilized in Section 4 of Chapter 2.

CHAPTER 2

A METHOD OF SPLITTING AN RC TRANSFER FUNCTION VECTOR BY CREATING DEGENERACIES

1. Introduction

Kodali[8] has developed a method of calculating the reduced transfer functions t' and t" which utilizes both the arbitrary splitting factor λ of equation (1.3) and the arbitrariness of the polynomial D_q. The method outlined below extends his results to vector transfer functions and explicitly considers functions of arbitrary degree. The method also provides useful values for the split factor λ.

As we saw in the last chapter, the actual method used to obtain the vectors $\underline{t}'$ and $\underline{t}''$ is arbitrary provided that the coefficient conditions are satisfied. The method presented below forces the numerator coefficients to satisfy the equalities of equation (1.2) as much as possible. This leads to fewer components being needed in the synthesis of the unity degree transfer functions.

2. Transfer Function Calculations

The method below considers the coefficients of s^{r-i} as a group of numbers which can be split into

two new groups such that the sum of one of the new groups is either zero, b', or b". One of the new groups is then assigned to $\underline{t}'$ and the other to $\underline{t}''$ as the numerator coefficients of s^{r-i-1} and s^{r-i} respectively. The reader can best follow the general method of calculating $\underline{t}'_q$ and $\underline{t}''_q$ by considering a particular case first.

Example 2.1. Suppose that the numerator coefficients of s^3 in a five entry transfer function vector of degree seven are given by $a_{41} = 3$, $a_{42} = 5$, $a_{43} = 2$, $a_{44} = 9$, and $a_{45} = 1$, or more compactly, as $\underline{a}_4 = (3, 5, 2, 9, 1)$, and that the denominator coefficient is $b_4 = 30$. Suppose also that in splitting the admittance by using equations (1.4) and (1.5) we obtain $b'_4 = 21$ and $b''_4 = 9$. Then using the proportional method of equation (1.8) for calculating the reduced transfer functions we obtain

$$\underline{a}'_4 = (\frac{21}{10}, \frac{7}{2}, \frac{7}{5}, \frac{63}{10}, \frac{7}{10}) \text{ and } \underline{a}''_4 = (\frac{9}{10}, \frac{3}{2}, \frac{3}{5}, \frac{27}{10}, \frac{3}{10}).$$

In splitting these coefficients we only require that equation (1.2) be valid, namely that

$$0 \leq \sum_{j=1}^{5} a'_{4j} \leq 21 \quad , \quad 0 \leq \sum_{j=1}^{5} a''_{4j} \leq 9 \quad ,$$

and each coefficient is non-negative, and that $a'_{4j} + a''_{4j} = a_{4j}$. It is apparent that these conditions are also met by each of the following:

i) $\underline{a}'_4 = (3, 5, 2, 0, 1)$, $\underline{a}''_4 = (0, 0, 0, 9, 0)$,

iia) $\underline{a}'_4 = (0, 0, 1, 9, 1)$, $\underline{a}''_4 = (3, 5, 1, 0, 0)$,

iib) $\underline{a}'_4 = (0, 0, 2, 9, 0)$, $\underline{a}''_4 = (3, 5, 0, 0, 1)$,

iii) $\underline{a}'_4 = (3, 5, 2, 9, 1)$, $\underline{a}''_4 = (0, 0, 0, 0, 0)$.

The advantage of this second type of calculation lies in the number of zero coefficients introduced, since, in the last stage of a synthesis, every non-zero coefficient in the augmented transfer function vector is proportional to the inverse value of a resistor or capacitor. In the following pages it will be shown that one can always calculate $\underline{a}'_4$ and $\underline{a}''_4$ in this manner and that the worst possible case, in terms of the number of non-zero coefficients, yields four instead of five zeros for a five-entry vector.

For the sake of clarity let us drop the subscripts q and i, i.e., let $\underline{t}_q = \underline{t}$ and let $a_{iqj}s^{r-i} = a_j s^{r-i}$.

The three basic splits of Example 2.1 correspond to the three methods that can be used to make the coefficients equal to 0, b', or b". Assuming that

$b' \leq b''$, the coefficients of s^{r-i} in the numerator and denominator of the vector $\underline{t}$ must conform to at least one of the following three classifications:

<u>Class i)</u> $a_k \geq b'$ for some k between 1 and m.

With $a_k \geq b'$ we can split a_k into two parts; the first will equal b'; the remainder and all the other coefficients will form $\underline{t}''$.

Putting this split in equation form we have

(2.1a) $a'_k = b'$ and $a''_k = a_k - b'$,

(2.1b) $a'_j = 0$ and $a''_j = a_j$, for all $j \neq k$.

It follows that such a split satisfies all of the coefficient conditions and that it creates (m-1) zero coefficients.

If $a_k \geq b''$ we may wish to use b'' and thus rewrite equation (2.1) with ' and '' interchanged.

<u>Class ii)</u> $a_j < b'$ for every j and yet $\sum_j a_j > b''$.

Since no one coefficient is large enough, a partial sum must be formed such that by adding a fraction of a_k to it, the

partial sum is equal to b". The remainder of a_k and the unused coefficients will form $\underline{t}'$. Thus we pick k such that

$$\sum_{j=1}^{k-1} a_j \leq b'' \text{ and } \sum_{j=1}^{k} a_j > b'' \tag{2.2}$$

and calculate the new coefficients by setting

$$a'_j = 0 \text{ and } a''_j = a_j \quad \text{for } j = 1, \ldots, k-1, \tag{2.3a}$$

$$a'_k = -b'' + \sum_{j=1}^{k} a_j \text{ and } a''_k = b'' - \sum_{j=1}^{k-1} a_j , \tag{2.3b}$$

$$a'_j = a_j \text{ and } a''_j = 0 \quad \text{for } j = k+1, \ldots, m. \tag{2.3c}$$

Since the entries of a transfer function vector can be moved about in the vector, one is free to group the coefficients in any manner. At least (m-1) coefficients will be zero but sometimes a careful grouping will lead to a sum which equals b", making a'_k zero (cf. Example 2.1 iib).

Since $b' \leq b''$, one could write a set of equations which assign a_k with respect to b'. These equations would merely be equations (2.2) and (2.3) with ' and " interchanged.

It is easily shown that the coefficient conditions are satisfied by equation (2.3): the a''_j were constructed to satisfy the conditions with respect to b"; summing the a'_j we obtain

$$\sum_{j=1}^{m} a'_j = (0) + (-b'' + \sum_{j=1}^{k} a_j) + (\sum_{j=k+1}^{m} a_j)$$

$$= \sum_{j=1}^{m} a_j - b'' \quad ,$$

but since $\sum_j a_j \leq b$ and since $b - b'' = b'$ it is apparent that the a'_j do indeed satisfy the coefficient conditions with respect to b'.

Class iii) $\qquad \sum_j a_j \leq b''$.

Under this condition there is no need to split coefficients since obviously the a_j already satisfy the coefficient conditions with respect to b" and thus can be put directly into t", i.e.,

(2.4) $\qquad a'_j = 0$ and $a''_j = a_j$ for all j.

This creates m zero coefficients.

Of course, if the sum is less than or equal to b' we are free to use equation (2.4) with ' and " interchanged.

Regardless of which of the above three methods is used, the coefficients of s^r and s^0 must be treated as in equation (1.7a).

When the transfer function is a scalar, the more convenient classifications are: 1) $a_k \leq b''$ to use method iii) and 2) $a_k > b''$ to use method i). The second vector classification for method ii) vanishes trivially.

3. <u>The Number of Components in a Synthesis</u>

To evaluate the effect of the above split on the number of components needed in a synthesis two definitions will be introduced.

<u>Definition 1</u>. <u>A numerator coefficient</u>, a_{ij}, <u>of</u> t <u>is said to be</u> degenerate <u>if it is zero</u>.

<u>Definition 2</u>. <u>The</u> degeneracy, δ_+, <u>of a transfer function vector</u> t <u>is the number of degenerate coefficients in its augmented vector</u> t^+.

In a unity degree augmented transfer function vector each non-zero coefficient is a component and thus each degeneracy is a missing component (see Figure 2.1). With this in mind, we will proceed to

calculate the number of degeneracies at the end of a synthesis. Once this number is known, determining the number of components is a trivial matter.

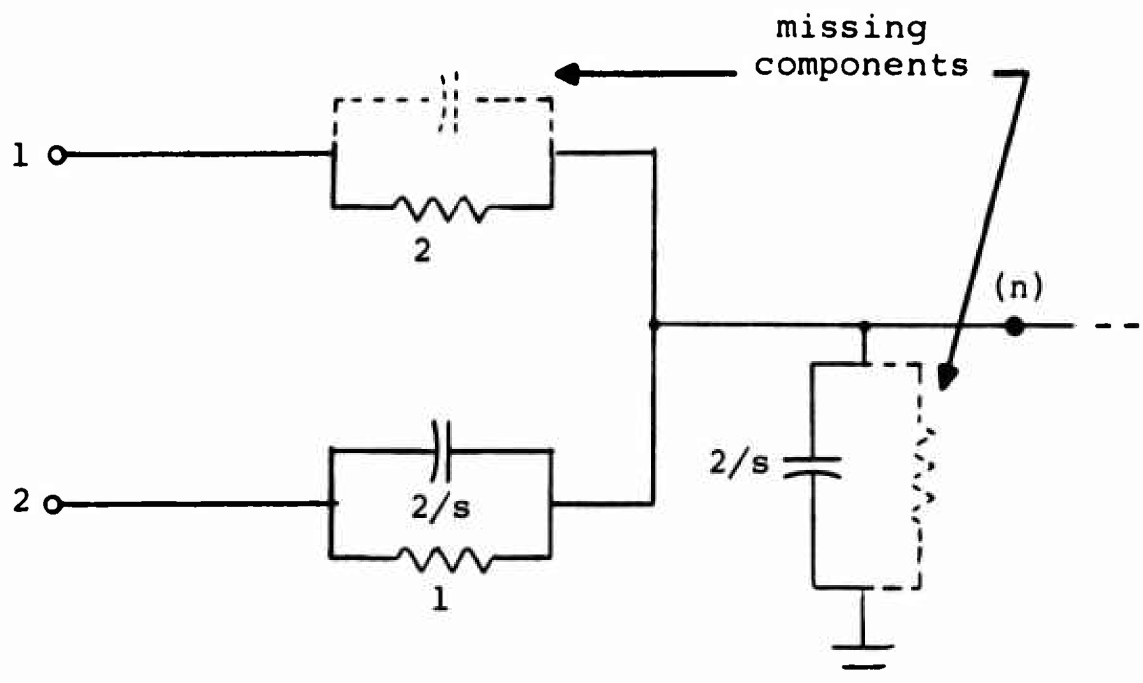


Fig. 2.1 Synthesis of $\underline{t}_+^{(n)} = \left[\frac{s}{2s+3}, \frac{1}{2s+3}, \frac{s+2}{2s+3}\right]$

with $Y^{(n)} = \frac{2s+3}{2}$ and $\delta_+^{(n)} = 2$.

In all of the discussion and theorems that follow it is assumed that one method of calculating the reduced transfer functions is used exclusively in a synthesis.

Theorem 3. The degeneracy of a transfer function vector of degree r with m entries satisfies $\delta_+ \leq m(r+1)$.

Proof. In the vector $\underline{t}^+$ there are (m+1)(r+1) numerator coefficients of which at least (r+1) are non-zero by equation (1.9). Hence there are at most m(r+1) degeneracies. To exemplify the extremes:

$$\delta_+ = 0 \quad \text{for } \underline{t} = \begin{bmatrix} s + 1 \\ 2s + 3 \end{bmatrix}$$

$$\delta_+ = 6 \quad \text{for } \underline{t} = \left[\frac{s^2}{s^2 + 3s + 1} , \frac{3s}{s^2 + 3s + 1} \right] .$$

QED

Notice that the degeneracy of a network is the sum of the degeneracies of the vectors which describe that network and as such it will, in general, increase as the synthesis proceeds.

Theorem 4. Using the degeneracy method of splitting t, the net increase in the degeneracy of a network is between m(r-1) and (m+1)(r-1) at each cycle, i.e.,

$$\delta_+^{(1)} + \delta_+^{(2)} = \delta_+ + d, \text{ where } m(r-1) \leq d \leq (m+1)(r-1).$$

Proof. Upon examining the degeneracy split we see that only (r-1) coefficients are involved in the calculation since the coefficients of s^r and s^0 are

not split. Each of these (r-1) coefficients introduces either m or (m+1) new degeneracies, depending upon whether or not the remainder is zero. Hence the asserted limits.

QED

Theorem 5. Using the degeneracy split, if $\delta_+ = \Delta$ for a vector t, then the final network will have $(\Delta+D)$ degeneracies, where

$$m(2^r - r-1) \leq D \leq (m+1)(2^r - r-1).$$

Proof. Using Theorem 4, after the first cycle

$$\delta_+^{(1)} + \delta_+^{(2)} = \Delta + d_0 \text{ , where } m(r-1) \leq d_0 \leq (m+1)(r-1).$$

For the second cycle

$$\delta_+^{(3)} + \delta_+^{(4)} + \delta_+^{(5)} + \delta_+^{(6)} = \Delta + d_0 + 2d_1 \quad ,$$

where $m(r-2) \leq d_1 \leq (m+1)(r-2)$ since r was decreased by one. We have used the notation of Figure 1.3 that $\underline{t}^{(p)}$ splits into $\underline{t}^{(2p+1)}$ and $\underline{t}^{(2p+2)}$.

Similar terms are added for the (r-1) times that the transfer function is split to reduce the degree and the final result is

$$\Delta + m(r-1) + 2m(r-2) + 4m(r-3) + \ldots + 2^{r-2}m$$

$$\leq \sum_{i=2^{r-1}-1}^{2^r-2} \delta_+^{(i)} \leq$$

$$\Delta + (m+1)(r-1) + 2(m+1)(r-2) + \ldots + 2^{r-2}(m+1) \quad ,$$

which is equivalent to

$$\Delta + m(2^r-r-1) \leq \sum_{i=2^{r-1}-1}^{2^r-2} \delta_+^{(i)} \leq \Delta + (m+1)(2^r-r-1).$$

<u>QED</u>

In Example 2.2 below the results of the computer program in the Appendix are used to show how the number of degeneracies increases as the degree of the transfer function vectors decreases.

<u>Example 2.2</u>. Suppose that the transfer function vector is given by

$$\underline{t}_3 = \left[\frac{8.88s^3+18s^2+17.76s}{s^4+9s^3+23s^2+18s+4}, \frac{s^4+5s^2+4}{s^4+9s^3+23s^2+18s+4}\right]$$

and the driving-point admittance by

$$Y_{33} = \frac{s^4 + 9s^3 + 23s^2 + 18s + 4}{4s^3 + 27s^2 + 46s + 18} .$$

Notice that $\delta_+ = 7$ so that by Theorem 5 there will be between 29 and 40 degeneracies in the final vectors. After the first cycle there are 13 degeneracies since

$$\underline{t}_3^{(1)} = \left[\frac{7.31s^2+9.25s+6.14}{s^3+7.43s^2+14.25s+6.14} , \frac{s^3+5s}{s^3+7.43s^2+14.25s+6.14} \right]$$

$$\underline{t}_3^{(2)} = \left[\frac{1.57s^3+8.75s^2+11.6s}{1.57s^3+8.75s^2+11.9s+4} , \frac{4}{1.57s^3+8.75s^2+11.9s+4} \right]$$

The second cycle (with $\lambda = 0.5$ everywhere except for splitting $Y_{33}^{(1)}$ of $\underline{t}_3^{(1)}$, when it is 0.387) gives 10 more degeneracies in the vectors

$$\underline{t}_3^{(3)} = \left[\frac{5.15s}{s^2+5.27s+5} , \frac{s^2+5}{s^2+5.27s+5} \right]$$

$$\underline{t}_3^{(4)} = \left[\frac{2.16s^2+9.25s+6.14}{2.16s^2+9.25s+6.14} , 0 \right]$$

$$\underline{t}_3^{(5)} = \left[\frac{1.57s^2+5.80s+3.19}{1.57s^2+5.80s+3.19} , 0 \right]$$

$$\underline{t}_3^{(6)} = \left[\frac{2.95s^2+8.43s}{2.95s^2+8.67s+4} , \frac{4}{2.95s^2+8.67s+4} \right].$$

The third, and final, cycle gives the following unity

degree transfer function vectors:

$$\underline{t}_3^{(7)} = \left[\frac{3.51}{s + 3.62} , \frac{s}{s + 3.62} \right]$$

$$\underline{t}_3^{(8)} = \left[\frac{1.64s}{1.64s + 5} , \frac{5}{1.64s + 5} \right]$$

$$\underline{t}_3^{(9)} = \left[\frac{2.16s + 4.93}{2.16s + 4.93} , 0 \right]$$

$$\underline{t}_3^{(10)} = \left[\frac{4.32s + 6.14}{4.32s + 6.14} , 0 \right]$$

$$\underline{t}_3^{(11)} = \left[\frac{1.57s + 3.47}{1.57s + 3.47} , 0 \right]$$

$$\underline{t}_3^{(12)} = \left[\frac{2.33s + 3.19}{2.33s + 3.19} , 0 \right]$$

$$\underline{t}_3^{(13)} = \left[\frac{2.95s + 2.69}{2.95s + 2.69} , 0 \right]$$

$$\underline{t}_3^{(14)} = \left[\frac{5.74s}{5.98s + 4} , \frac{4}{5.98s+4} \right]$$

The network which synthesizes $\underline{t}_3$ is given in Figure 2.2 and since the final vectors have a total of 30 degeneracies there are only 32 components in the network.

Returning to the task of calculating the number of components in a network we have:

<u>Theorem 6.</u> <u>The maximum number of components needed in a Fialkow-Gerst synthesis employing only the de-</u>

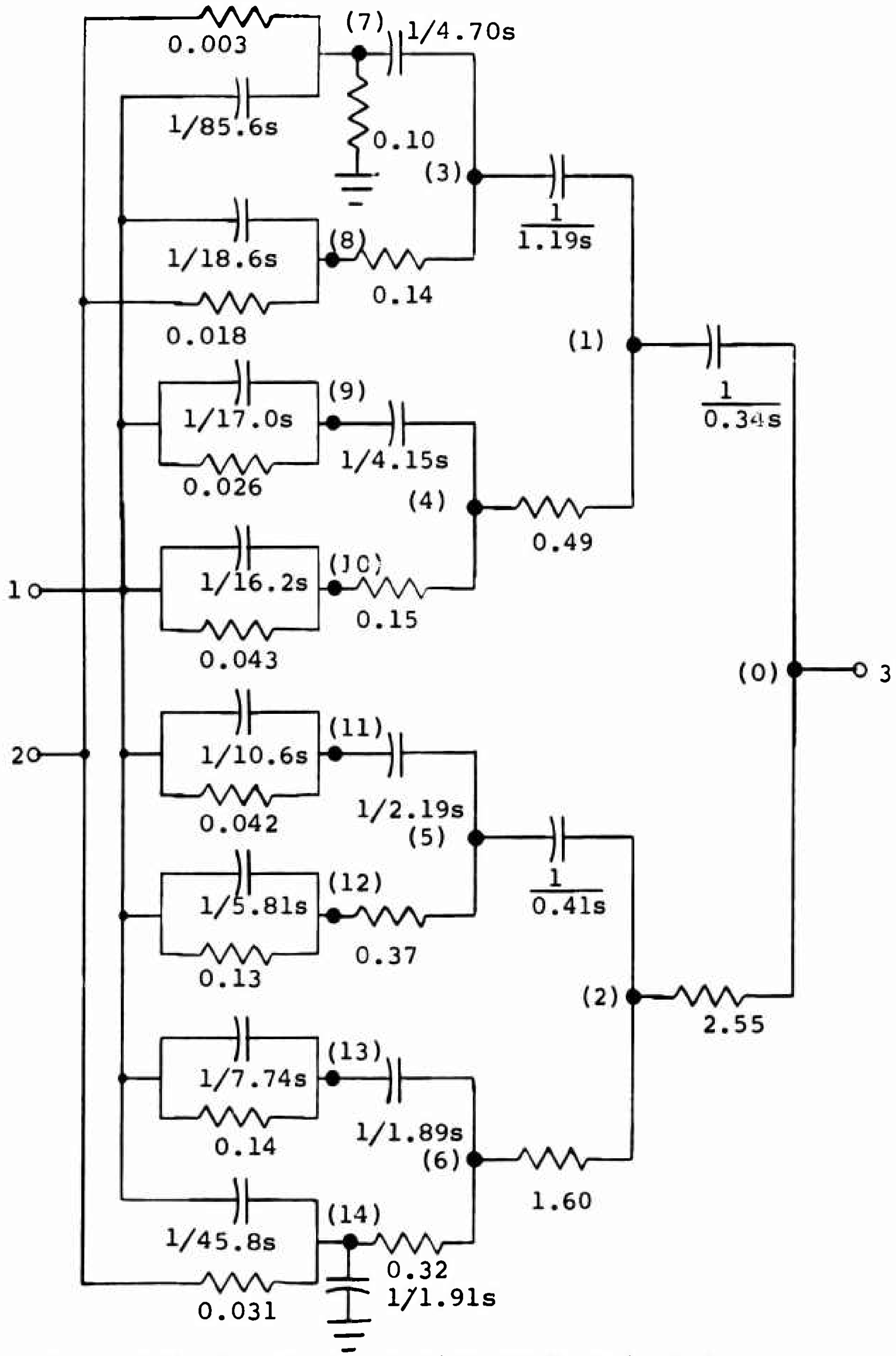


Fig. 2.2 Computer solution of Example 2.2.

<u>generacy split is $(2^{r+1}-2+m(r+1))$</u>.

<u>Proof</u>. Considering the absolute worst case of D = 0, by Theorem 5 there are at least $m(2^r-r-1)$ degeneracies when the synthesis is complete. Since there are $2^r(m+1)$ coefficients among the unity degree transfer functions, the difference, or $2^r+m(r+1)$, will appear as components in the network termination. Adding to this number the (2^r-2) components in the network tree we obtain the asserted result of $2^{r+1}-2+m(r+1)$ components.

A particular transfer function vector may use fewer components or a practical realization may use more but $(2^{r+1}-2+m(r+1))$ is the " maximum number of components needed", i.e., it is a sufficient number.

<u>Corollary 7.</u> <u>The maximum number of components needed in a Fialkow-Gerst synthesis employing only the proportional split is $(2^{r+1}-2+m2^r)$</u>.

<u>Proof.</u> It is evident that if the transfer function vector has no degeneracies at the start then the proportional split will not introduce any. Hence none will appear in the unity degree transfer function vectors and all the termination components will be present. These termination components and the tree components add to $(2^{r+1}-2+m2^r)$.

<u>QED</u>

The major difference between the two methods is the factor of (r+1) instead of 2^r in the maximum. The difference between the methods in terms of the minimum number of components obtainable is not as dramatic. In fact, the real difference lies not in the number of components but in the number of inputs, or vector entries, that each method can tolerate before the absolute minimum of $(2^{r+1}-2)$ becomes unobtainable. It will also be seen that the degeneracy split permits a larger variety of minimal vector forms.

Lemma 8. The minimum number of components obtainable using the proportional split for $m \leq (r+1)$ is $(2^{r+1}-2)$ and for $m > (r+1)$ is $(2^{r+1}-2) + (m-r-1)$.

Proof. Regardless of the method of calculation, 2^{r-1} unity degree transfer function vectors will be produced in a synthesis. Since each of these has at least two non-zero coefficients in its augmented vector there must be at least 2^r components in the network termination. When the components in the network tree are added there is a total of $(2^{r+1}-2)$ components.

Since a maximally degenerate vector will use a minimum number of components the problem becomes: when is it impossible to use a maximally degenerate vector? For the proportional split this cutoff occurs when all the denominator coefficients have

been used to form entries, i.e., when $m > (r+1)$. If the extra entries are created by using fractions of the coefficients of s^0 or s^r there will be an increase of only one component for each of the fractional entries. This is due to the fact that the coefficients of s^0 and s^r are not split in the synthesis.

The existence of the vectors is demonstrated by

$$\underline{t} \left[\frac{s^2}{2s^2+3s+1}, \frac{3s}{2s^2+3s+1}, \frac{1}{2s^2+3s+1}, \frac{s^2}{2s^2+3s+1}\right]$$

which only requires seven components.

<u>QED</u>

It should be noted that if any other coefficient is used then, as the vector is split, this extra coefficient will appear in more and more vectors until it becomes the s^r or s^0 coefficient. These excess coefficients will of course produce excess components.

<u>Theorem 9</u>. <u>The minimum number of components obtainable using only the degeneracy split for</u> $m \leq 2^r$ <u>is</u> <u>$(2^{r+1}-2)$ and for</u> $m > 2^r$ <u>is $(2^{r+1}-2) + (m-2^r)$</u>.

<u>Proof</u>. The calculation is the same as in Lemma 8 and the transfer function vector forms are essentially the same. The entry limit of 2^r comes from the fact that a number which is in the original vector numerator

can be carried, unaltered, to one of the unity degree transfer function vectors. Thus, corresponding to Example 2.2, by starting with the vector

$$\underline{t}_3 = \frac{1}{s^4+9s^3+23s^2+18s+4}\left[s^4,\ 1.57s^3\ ,\ 2.95s^2,\ 2.33s^2\ \ 3.47s^2\ ,\ 6.14s\ ,\ 4\right]$$

one would get terminal vectors such as

$$\underline{t}_3^{(7)} = \left[\frac{s}{s+3.62}\ ,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0\right]$$

$$\underline{t}_3^{(11)} = \left[0,\ \frac{1.57s}{1.57s+3.47}\ ,\ 0,\ 0,\ \frac{3.47}{s+3.47}\ ,\ 0,\ 0\right]$$

and this network would require only the minimum thirty components. Obviously the limit is the number of denominator coefficients, 2^r, that are available in the termination vectors, not the $(r+1)$ in the original vector, as in the proportional split.

When $m > 2^r$ fractions of coefficients may be used to obtain a minimum of components but they are no longer restricted to any particular coefficient.

<u>QED</u>

Applying Theorems 6 and 9 to Example 2.2 the synthesis will yield between thirty and forty components for the degeneracy method and between thirty and

sixty-two for the proportional method. After synthesizing it both ways, it is found that the proportional split synthesis used forty-four components and the degeneracy split used thirty-two. In the next section it will be shown that even the latter number can be reduced.

4. Hybrid Synthesis

The degeneracy split greatly enhances the possibility of obtaining special transfer functions which can be synthesized by non-Fialkow-Gerst methods, at a savings of components. The example below illustrates this point.

Example 2.3. Examining the transfer functions of Example 2.2 we see that both $\underline{t}_3^{(4)}$ and $\underline{t}_3^{(5)}$ are trivial functions of the form $\underline{t} = [1,0]$, with second degree driving-point admittances given by the computer as

$$Y_{33}^{(4)} = \frac{2.16s^2+9.25s+6.14}{0.65s+1.19}$$

and

$$Y_{33}^{(5)} = \frac{1.57s^2+5.80s+3.19}{0.86s+1.58} .$$

Such transfer functions can be synthesized as in Figure 2.3 by a ladder network. When this is done for the problem at hand we obtain Figure 2.4, which

has only twenty-eight components--two less than the absolute minimum that is obtainable with the pure Fialkow-Gerst synthesis.

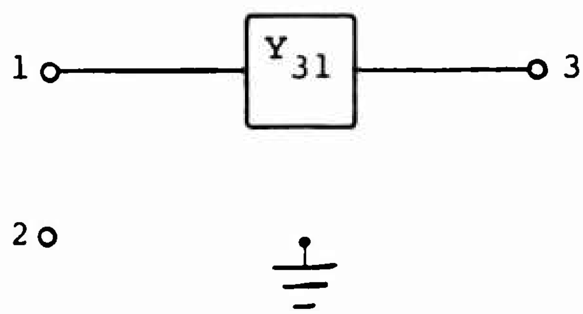


Fig. 2.3 Synthesis of $\underline{t}$ = [1,0].

5. Optimization of the Degeneracy Split Using the Split Factor λ

The degeneracy split is based upon the idea of splitting a numerator coefficient such that the new coefficients equal 0, b' , or b". In general, this leaves a non-zero remainder for the other vector. Let us now attack the problem from the other direction by forcing the denominator coefficients to be equal to the numerator coefficients.

Recall that the new vector denominators are calculated in equation (1.3) when the admittance is split. Recall also, that in splitting the admittance we use a split factor λ which is arbitrary but restricted to the range of numbers between zero and

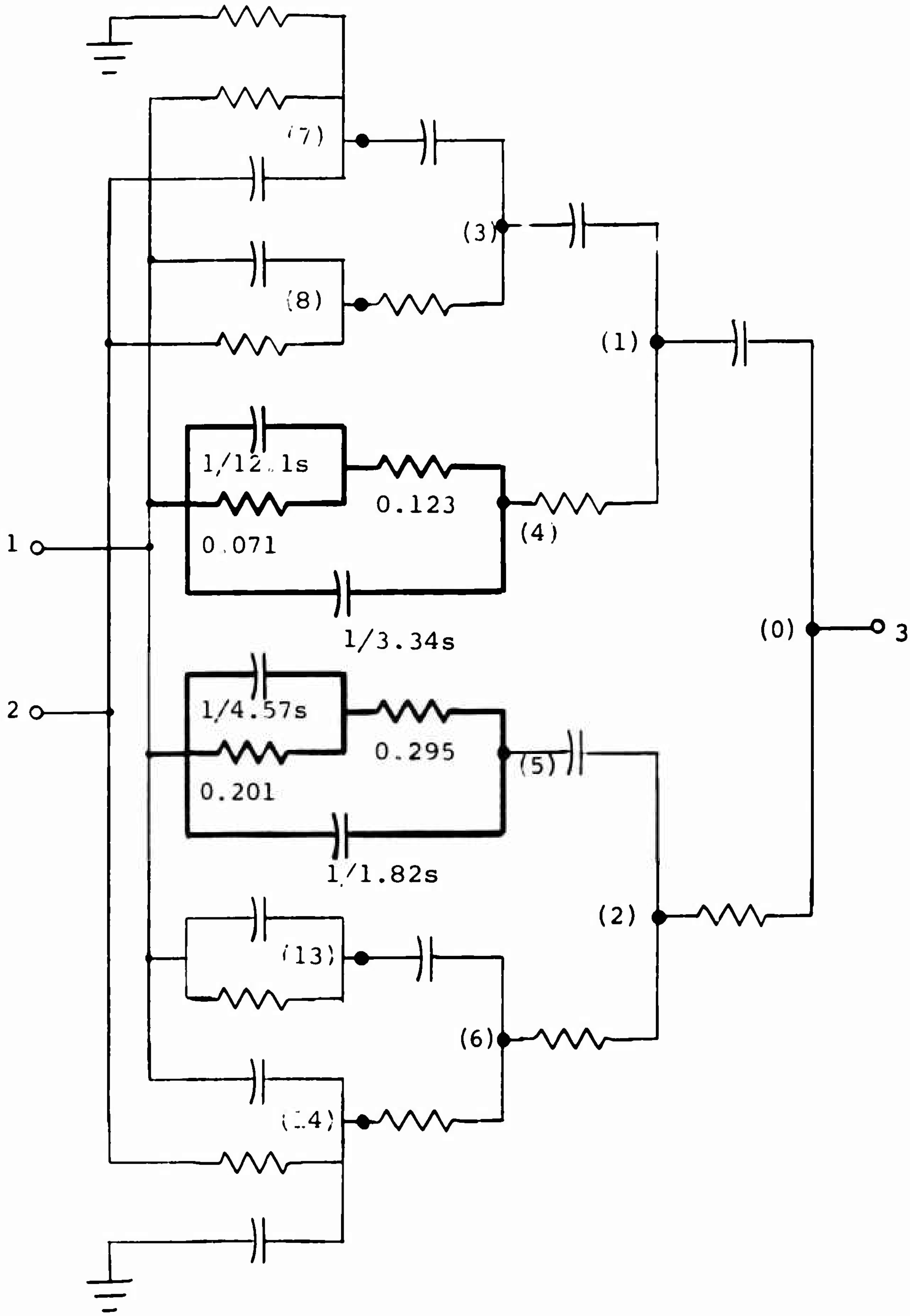


Fig. 2.4 Hybrid synthesis of Example 2.2 (modification of Fig. 2.2).

one.* Since equations (1.4c) and (1.5b) relate b_i to λ they can be used to calculate λ for given values of b_i.

Several values of b_i are useful in that they increase the degeneracy of the transfer function vector. For example, in equation (2.1) we would like the remainder a''_{ik} to be zero, hence $a''_{ik} = a_{ik} - b'_i = 0$. If $a''_{ik} = a^{(1)}_{ik} = 0$ then equation (1.4c) will yield

$$(2.5a) \qquad \lambda = \frac{a_{ik} - b_0 d_i/d_0}{b_i - b_0 d_i/d_0 - b_r d_{i-1}/d_{r-1}} \; .$$

If $a''_{ik} = a^{(2)}_{ik} = 0$ then equation (1.5b) gives

$$(2.5b) \qquad \lambda = \frac{a_{ik} + b_0 d_i/d_0 - b_i}{b_i - b_0 d_i/d_0 - b_r d_{i-1}/d_{r-1}} \; .$$

Similarly in equation (2.4) we would like a'_j to be zero and this can be done by using equation (2.5) with $\sum_j a_{ij}$ substituted for a_{ik}. Equation (2.3) can also be used by substituting $\sum_{j=1}^{k-1} a_{ij}$ for a_{ik} in equation (2.5).

---

*Kodali[8] has investigated the conditions that an RC transfer function must meet if λ equals zero or one.

6. Summary and Conclusion

In this chapter we have shown the advantage of the degeneracy split over the proportional split.. Example 2.3 showed one of these, namely the creation of easily realized transfer functions. This use of the degeneracy split appears to be promising and will be the subject of future investigations.

Secondly, the use of a calculated value of the split factor λ is important since the introduction of even one degeneracy early in the synthesis of a vector leads to many additional degeneracies in the unity degree transfer functions.

Finally, the most significant contribution of the degeneracy split is that it can be used on any RC transfer function vector, of any degree, and that it greatly reduces the number of components needed to synthesize that vector. Table 2.1 shows how large this difference can become for transfer functions of low degree. The fact that these maxima were predicted by counting a particular type of coefficient suggests that other properties of networks may also be predicted from the original coefficients.

M_d/M_p

r / m	2	3	4	5	...	10
1	9/10	18/22	35/46	68/94		2057/3070
2	12/14	22/30	40/62	74/126		2068/4094
3	15/18	26/38	45/78	80/158		2079/5118
4	18/22	30/46	50/94	86/190		2090/6142
5	21/26	34/54	55/100	92/222		2101/7166

Table 2.1 A comparison of the maximum number of components in a synthesis using the degeneracy split, M_d, or the proportional split, M_p.

Appendix

COMPUTER SYNTHESIS

A computer program has been written for synthesizing an RC transfer function vector and its driving-point admittance using the Fialkow-Gerst method. The program incorporates several of the techniques introduced in Chapter 2 and allows for hybrid synthesis by printing all the calculated vector and admittance coefficients. For networks with only one or two inputs, a subprogram is available for scaling the components to a given frequency and for placing the component values in their appropriate place in a printed circuit board.

A general flow chart of the program is given in Figure A.1 and it is assumed that the reader is sufficiently familiar with ALGOL 60 that the details will be evident from the program itself. The one notational deviation from the text arises from a programming problem---when the admittances and transfer function vectors are split, the even numbered parts have subscripts which differ from the text by unity, thus A(I,P,J) = A(I,2P+1,J)+A(I-1,2P+2,J) instead of $a_{ij} = a'_{ij} + a''_{ij}$.

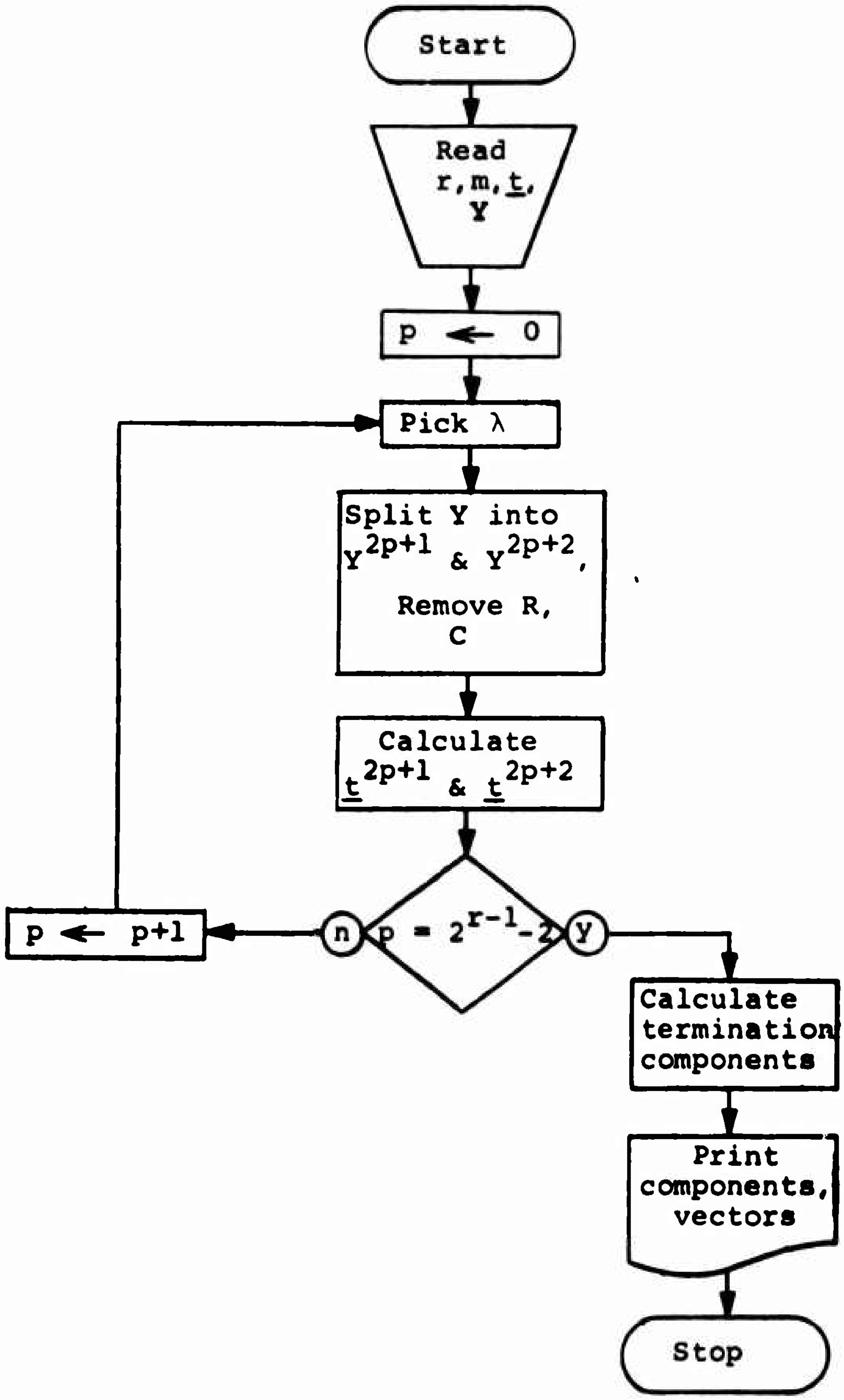


Fig. A.1 General flow chart.

The great flexibility of most synthesis procedures creates problems when an attempt is made to write a program for them and the Fialkow-Gerst is no exception. Some of the design decisions are mentioned below so that the reader will be aware of the program's shortcomings.

When several split factors are available for a vector the program picks the one nearest to 0.5 in order to minimize the range of component values. The program makes no attempt to consider future vectors.

Using the chosen λ, the admittance is split and the resistor and capacitor are removed. When the transfer function vectors $\underline{t}^{(2p+1)}$ and $\underline{t}^{(2p+2)}$ are calculated a set sequence is followed. If method i) is used then the largest numerator coefficient, MXA, is used as a_k and the smallest denominator, MNB, is used as b' in equation (2.1). This was done to minimize the range of component values. If method ii) is used, no special grouping or testing is done---the coefficients are summed in order, starting with entry 1.

For these and other reasons the program is sub-optimum and it is suggested that the user try all possible permutations of the vector entries in the input data, including using the complementary term

as an input. Using the complementary transfer function as an input will also emphasize the rounding errors of the computer which appear as grounded "components" with values of $\pm 10^{\pm 7}$.

The compilation of the program requires about 4500 memory locations, excluding the array storage.

```
                                                               COMMENT
THIS ALGOL 60 PROGRAM WILL PRODUCE A FIALKOW-GERST SYNTHESIS RE
FIALKOW ET AL. IEEE TRAN. CKT THEORY, VOL CT-11 NO 1 MARCH 1964,
FROM A GIVEN RC TRANSFER FUNCTION VECTOR AND ITS DRIVING POINT
ADMITTANCE.   THE RESISTORS REMOVED ARE NUMBERED SEQUENTIALLY.
THUS NODE 0 IS 0, NODE 1 IS 1, 2 IS 2, 11 IS 3, 12 IS 4, 21 IS 5,
22 IS 6, 111 IS NUMBERED 7, ETC.   THE SUBSCRIPT 0 REFERS TO GROUND.
                                                               $
                                                                         COMMENT  $
REAL RET,SUM,MXA,MNR,MXR,DIFF,BESTLAM$

REAL LAM,DEL,SUMY,SUMC,DENOM,NUMER1,NUMER2$
INTEGER U,P,V,R,M,I,J,K,L,X,Y,TOP,DONE, DEGREE,NUMINPUTS,MNP,MXP,MXK$
FORMAT FRMT0('DEGREE=',I3,X5,'NO. INPUTS=',I3,A1.0)$
FORMAT FRMT1( 'THE DENOMINATOR COEFFICIENTS ARE',A2.0,
     (5R14.8,A1.0))$
FORMAT FRMT2('THE NUMERATOR COEFFICIENTS OF ENTRY',I3,' ARE',A2.0,
     (5R14.8,A1.0))$
FORMAT FRMT3('THE NUMERATOR COEFFICIENTS OF Y(0,0) ARE',A2.0,
     (5R14.8,A1.0))$
FORMAT FRMT4('THE DENOMINATOR COEFFICIENTS OF Y(0,0) ARE',A2.0,
     (5R14.8,A1.0))$
FORMAT FRMT5(I3,X2,I3,X8,'NONE',X13,'NONE',A1.0)$
FORMAT FRMT6(I3,X2,I3,X3,R14.8,X8,'NONE',A1.0)$
FORMAT FRMT7(I3,X2,I3,X8,'NONE',X8,R14.8,A1.0)$
FORMAT FRMT8(I3,X2,I3,X3,R14.8,X3,R14.8,A1.0)$
FORMAT FRMT9('THE FOLLOWING COMPONENTS ARE IN FARADS AND OHMS',A2.0,
     'THE FREQUENCY IS ONE RADIAN/SECOND',A1.0,
     'THE TREE COMPONENTS ARE',A1.0,
     'NODE',X8,'CAP',X13,'NODE',X8,'RES',A1.0)$
FORMAT FRMT10(I3,X3,R14.8,X8,I3,X3,R14.8,A1.0)$
FORMAT FRMT11('THE TERMINATING COMPONENTS ARE',A2.0,
     'NODE INPUT',X6,'CAP',X14,'RES',A1.0)$
FORMAT FRMT12(E0)$
FORMAT FRMT13('VECTOR NUMERATOR COEFFICIENT MATRIX',A2.0,
     'NODE INPUT',A1.0)$
FORMAT FRMT14(I3,X3,I3,X4,(9R12.5,A1.0))$
FORMAT FRMT15(X6,I3,X4,(9R12.5,A1.0))$
FORMAT FRMT16('VECTOR DENOMINATOR COEFFICIENT MATRIX',A2.0,
     X6,'NODE',A1.0)$
FORMAT FRMT17('DRIVING POINT ADMITTANCE DENOMINATOR COEFFICIENT MATRIX',
     A2.0,X6,'NODE',A1.0)$
                                                          COMMENT $
STRT..READ(DEGREE,NUMINPUTS)$
WRITE(FRMT12)$
R=DEGREE$
M=NUMINPUTS$
TOP=(2**R)-2$
DONE=(2**(R-1))-2$
BEGIN
REAL ARRAY A(0..R,0..TOP,0..M), B,D(0..R,0..TOP),CAP,RES(0..TOP),

ZY,ZC(DONE..TOP,0..M)$
                                                               COMMENT
READ DATA AND PRINT IT
                                                               $
READ(FOR J=(1,1,M) DO FOR K=(0,1,R) DO A(K,0,J),
      FOR J=(0,1,R) DO B(J,0),
      FOR J=(0,1,R-1) DO D(J,0))$
WRITE(FRMT1,FOR K=(0,1,R) DO B(K,0))$
FOR J=(1,1,M) DO WRITE(FRMT2,J,FOR K=(0,1,R) DO A(K,0,J))$
WRITE(FRMT3,FOR K=(0,1,R) DO B(K,0))$
WRITE(FRMT4,FOR K=(0,1,R-1) DO D(K,0))$
                                                               COMMENT
START THE SYNTHESIS
                                                               $
FOR U=(0,U+2,DONE) DO BEGIN FOR P=(U/2+1,U) DO BEGIN
FOR I=(0,1,R) DO FOR J=(1,1,M) DO A(I,P,0)=A(I,P,0)+A(I,P,J)$
```

```
                                                              COMMENT
CALCULATE POSSIBLE OPTIMIZING LAMBDA AND CHOOSE CLOSSEST ONE TO 1/2
                                                               $
FOR X=(1,1,R-1) DO FOR Y=(0,1,M) DO BEGIN
BESTLAM=0 $
DENOM=B(X,P)-B(0,P)*D(X,P)/D(0,P)-B(R,P)*D(X-1,P)/D(R-1,P)$
NUMER1=A(X,P,Y)-B(0,P)*D(X,P)/D(0,P)$
NUMER2=A(X,P,Y)+B(0,P)*D(X,P)/D(0,P)-D(X,P)$
IF DENOM EQL 0 THEN GO TO JUMPOUT $
LAM=NUMER1/DENOM $
DEL=-NUMER2/DENOM $
IF (LAM GEQ 1) OR (LAM LEQ 0) THEN LAM =0 $
IF (DEL GEQ 1) OR (DEL LEQ 0) THEN DEL =0 $
IF ABS(DEL-0.5) LSS ABS(LAM-0.5) THEN LAM=DEL $
IF ABS(LAM-0.5) LSS ABS(BESTLAM-0.5) THEN BESTLAM=LAM $
JUMPOUT..
END PICK OF BEST LAMBDA .
IF BESTLAM EQL 0 THEN BESTLAM =0.5 $
                                                              COMMENT
REMOVE RC COMPONENTS AND CAL     E NEW ADMITTANCES
                                                              $
LAM=BESTLAM $
DEL=1-BESTLAM $
B(0,2*P+1)=B(0,P)$
FOR I=(1,1,R-1) DO B(I,2*P+1)=B(I,P)*LAM
  +(B(0,P)*D(I,P)*DEL/D(0,P)) - (B(R,P)*D(I-1,P)*LAM/D(R-1,P))$
CAP(2*P+1)=B(R-1,2*P+1)/D(R-1,P)$
FOR I=(0,1,R-2) DO D(I,2*P+1)=D(I,P)-B(I,2*P+1)/CAP(2*P+1) $
FOR I=(0,1,R-2) DO B(I,2*P+2)=B(I+1,P)*DEL
  + B(R,P)*D(I,P)*LAM/D(R-1,P) - B(0,P)*D(I+1,P)*DEL/D(0,P)$
B(R-1,2*P+2)=B(R,P)$
RES(2*P+2)=D(0,P)/B(0,2*P+2)$
FOR I=(0,1,R-2) DO D(I,2*P+2)=D(I+1,P)-RES(2*P+2)*B(I+1,2*P+2) $
                                                           COMMENT
CALCULATE NEW TRANSFER FUNCTION NUMERATOR
                                                              $
FOR I=(1,1,R-1) DO BEGIN
MXA=0 $
FOR K=(1,1,M) DO MXA=MAX(MXA,A(I,P,K))$
FOR K=(1,1,M) DO IF A(I,P,K) EQL MXA THEN MXK=K $
MXB=MAX(B(I,2*P+1),B(I-1,2*P+2))$
MNB=MIN(B(I,2*P+1),B(I-1,2*P+2))$
IF MXB EQL B(I-1,2*P+2) THEN MXP=2*P+2 ELSE MXP = 2*P+1 $
IF MNB EQL B(I-1,2*P+2) THEN MNP=2*P+2 ELSE MNP = 2*P+1 $
SUM=A(I,P,0) $
IF SUM LEQ MNB THEN BEGIN
      FOR K=(1,1,M) DO BEGIN A(I,MNP,K)=A(I,P,K)$ A(I,MXP,K)=0 $ END$
      GO TO XFERDONE $          END$
IF SUM LEQ MXB THEN BEGIN
      FOR K=(1,1,M) DO BEGIN A(I,MXP,K)=A(I,P,K)$ A(I,MNP,K)=0 $ END$
      GO TO XFERDONE $          END$
FOR J=(1,1,M) DO BEGIN
      IF A(I,P,J) EQL MNB THEN BEGIN
      FOR K=(1,1,M) DO BEGIN A(I,MXP,K)=A(I,P,K)$ A(I,MNP,K)=0 $ END$
      A(I,MNP,J)=MNB $
      A(I,MXP,J)=0 $
      GO TO XFERDONE $          END$
      IF A(I,P,J) EQL MXB THEN BEGIN
      FOR K=(1,1,M) DO BEGIN A(I,MNP,K)=A(I,P,K)$ A(I,MXP,K)=0 $ END$
      A(I,MXP,J)=MXB $
      A(I,MNP,J)=0 $
      GO TO XFERDONE $          END$
                  END$
IF MXA GTR MNB THEN BEGIN
      FOR K=(1,1,M) DO BEGIN A(I,MXP,K)=A(I,P,K)$ A(I,MNP,K)=0 $ END$
      A(I,MNP,MXK) = MNB $
      A(I,MXP,MXK)=MXA-MNB $
      GO TO XFERDONE$           END$
```

```
IF (SUM GTR MXR) AND (MXR LSS MNR) THEN BEGIN
      SUM=0$
      FOR K=(1,1,M) DO BEGIN
           IF (SUM + A(I,P,K)) GEQ MNR THEN BEGIN
           DIFF=MNR - SUM $
           FOR J=(1,1,K-1) DO BEGIN A(I,MNP,J)=A(I,P,J)$A(I,MXP,J)=0$ENDS
           A(I,MNP,K) = DIFF $
           A(I,MXP,K)=A(I,P,K) - DIFF $
           FOR J=(K+1,1,M) DO BEGIN A(I,MXP,J)=A(I,P,J)$A(I,MNP,J)=0$ENDS
           GO TO XFERDONE $                            ENDS
           SUM = SUM + A(I,P,K) $
                                                   END $
                                                   END $
XFERDONE..
END TRANSFER FUNCTION CALCULATION $
FOR K=(1,1,M) DO FOR I=(1,1,R) DO A(I-1,2*P+2,K)=A(I,2*P+2,K) $
FOR K=(1,1,M) DO BEGIN A(0,2*P+1,K)=A(0,P,K)$
     A(R-1,2*P+2,K)=A(R,P,K)$ ENDS
END P $
R=R-1 $
END U, REMOVAL OF RC TREE $
FINALSYNTHESIS..
WRITE(FRMT9)$
FOR P=(0,1,DONE) DO WRITE(FRMT10,2*P+1,CAP(2*P+1),2*P+2,RES(2*P+2))$
                                                                      COMMENT
FINISH SYNTHESIS NOW THAT TRANSFER FUNCTIONS ARE OF UNITY DEGREE
                                                                      $
WRITE(FRMT11)$
FOR U=(DONE+1,1,TOP) DO BEGIN
SUMY=0$
SUMC=0$
FOR J=(1,1,M) DO SUMY=SUMY + A(1,U,J)$
FOR J=(1,1,M) DO SUMC=SUMC + A(0,U,J)$
ZY(U,0)=(B(1,U)-SUMY)/D(U,U) $
ZC(U,0)=(B(0,U)-SUMC)/D(U,U) $
FOR J=(1,1,M) DO ZY(U,J)=A(1,U,J)/D(0,U) $
FOR J=(1,1,M) DO ZC(U,J)=A(0,U,J)/D(U,U) $
FOR J=(0,1,M) DO BEGIN
     IF ZY(U,J) EQL 0 THEN BEGIN
          IF ZC(U,J) EQL 0 THEN BEGIN
               WRITE(FRMT5,U,J)$ GO TO NEXT $
                                      END $
          WRITE(FRMT6,U,J,ZC(U,J))$ GO TO NEXT
                                      ENDS
     IF ZC(U,J) EQL 0 THEN BEGIN
          WRITE(FRMT7,U,J, 1/ZY(U,J))$
          GO TO NEXT         ENDS
     WRITE(FRMT8,U,J,ZC(U,J), 1/ZY(U,J))$
NEXT..               END $
END TRANSFER SYNTHESIS $
WRITE(FRMT12)$
                                                                      COMMENT
INSERT PRINTED CIRCUIT BOARD PRINTOUT SUBPROGRAM HERE
                                                                      $

                                                                      COMMENT
PRINT OUT COEFFICIENT MATRICES
                                                                      $
R=DEGREE  $
WRITE(FRMT12)$
WRITE(FRMT13)$
FOR U=(0,U+2,TOP) DO BEGIN
     FOR P=(U/2,1,U) DO BEGIN
          K=1 $
          WRITE(FRMT14,P,K,FOR I=(U,1,R) DO A(I,P,K))$
          FOR K=(2,1,M) DO WRITE(FRMT15,K,FOR I=(0,1,R) DO A(I,P,K))$
                         ENDS
     R=R-1 $
                    END
```

```
R=DEGREE $
WRITE(FRMT12)$
WRITE(FRMT16)$
FOR U=(0,U+2,TOP) DO BEGIN
      FOR P=(U/2,1,U) DO WRITE(FRMT15,P,FOR I=(0,1,R) DO A(I,P))$
      R=R-1 $
                        ENDS
R=DEGREE $
WRITE(FRMT12)$
WRITE(FRMT17)$
FOR U=(0,U+2,TOP) DO BEGIN
      FOR P=(U/2,1,U) DO WRITE(FRMT15,P,FOR I=(0,1,R-1) DO D(I,P))$
      R=R-1 $
                        ENDS
WRITE(FRMT12)$
END WHOLE THING $

                                                         COMMENT
THIS COMPLETES THE SYNTHESIS OF ONE VECTOR OF THE MATRIX.
GO BACK AND DO THE NEXT VECTOR.
                                                         $
GO TO STRT $
FINISH $
```

```
THE DENOMINATOR COEFFICIENTS ARE
 0.9999999, 00 9.0000000, 00 2.2999999, 01 1.8000000, 01 4.0000000, 00

THE NUMERATOR COEFFICIENTS OF ENTRY  1 ARE
 0.0000000, 00 8.8799999, 00 1.8000000, 01 1.7759999, 01 0.0000000, 00

THE NUMERATOR COEFFICIENTS OF ENTRY  2 ARE
 0.9999999, 00 0.0000000, 00 5.0000000, 00 0.0000000, 00 4.0000000, 00

THE NUMERATOR COEFFICIENTS OF Y(Q,Q) ARE
 0.9999999, 00 9.0000000, 00 2.2999999, 01 1.8000000, 01 4.0000000, 00

THE DENOMINATOR COEFFICIENTS OF Y(Q,Q) ARE
 4.0000000, 00 2.7000000, 01 4.6000000, 01 1.8000000, 01

THE FOLLOWING COMPONENTS ARE IN FARADS AND OHMS
THE FREQUENCY IS ONE RADIAN/SECOND
THE TREE COMPONENTS ARE
```

NODE	CAP	NODE	RES
1	3.4104938,-01	2	2.5486725, 00
3	1.1856221, 00	4	4.9349726,-01
5	4.0910304,-01	6	1.5952959, 00
7	4.7002884, 00	8	1.3644900,-01
9	4.1546235, 00	10	1.5010854,-01
11	2.1936721, 00	12	3.6960558,-01
13	1.8865955, 00	14	3.2468323,-01

THE TERMINATING COMPONENTS ARE

NODE	INPUT	CAP	RES
7	0	NONE	9.7345446,-02
7	1	NONE	3.3358022,-03
7	2	8.5605738, 01	NONE
8	0	NONE	NONE
8	1	1.8624446, 01	NONE
8	2	NONE	1.7662571,-02
9	0	NONE	1.0651965, 06
9	1	1.7041002, 01	2.5734298,-02
9	2	NONE	NONE
10	0	NONE	NONE
10	1	1.0213461, 01	4.3358092,-02
10	2	NONE	NONE
11	0	NONE	-2.4724169, 06
11	1	1.0649865, 01	4.2469179,-02
11	2	NONE	NONE
12	0	NONE	NONE
12	1	5.8127440, 00	1.2576858,-01
12	2	NONE	NONE
13	0	NONE	NONE
13	1	7.7382428, 00	1.4167890,-01
13	2	NONE	NONE
14	0	1.9139785, 00	NONE
14	1	4.5785470, 01	NONE
14	2	NONE	3.1348291,-02

VECTOR NUMERATOR COEFFICIENT MATRIX

NODE	INPUT					
0	1	0.0000, 00	9.9709, 00	1.8000, 01	1.7759, 01	0.0000, 00
	2	0.9499, 00	0.0000, 00	5.0000, 00	0.0000, 00	4.0000, 00
1	1	0.0000, 00	7.3105, 00	9.2500, 00	6.1388, 00	
	2	0.9999, 00	0.0000, 00	5.0000, 00	0.0000, 00	
2	1	1.5694, 00	8.7499, 00	1.1621, 01	0.0000, 00	
	2	0.0000, 00	0.0000, 00	0.0000, 00	4.0000, 00	
3	1	0.0000, 00	5.1466, 00	0.0000, 00		
	2	0.9999, 00	0.0000, 00	5.0000, 00		
4	1	2.1638, 00	9.2500, 00	6.1388, 00		
	2	0.0000, 00	0.0000, 00	0.0000, 00		
5	1	1.5694, 00	5.9043, 00	3.1931, 00		
	2	0.0000, 00	0.0000, 00	0.0000, 00		
6	1	2.9456, 00	8.4279, 00	0.0000, 00		
	2	0.0000, 00	0.0000, 00	4.0000, 00		
7	1	0.0000, 00	3.5018, 00			
	2	0.9999, 00	0.0000, 00			
8	1	1.6448, 00	0.0000, 00			
	2	0.0000, 00	5.0000, 00			
9	1	2.1638, 00	4.9343, 00			
	2	0.0000, 00	0.0000, 00			
10	1	4.3156, 00	6.1388, 00			
	2	0.0000, 00	0.0000, 00			
11	1	1.5694, 00	3.4699, 00			
	2	0.0000, 00	0.0000, 00			
12	1	2.3344, 00	3.1931, 00			
	2	0.0000, 00	0.0000, 00			
13	1	2.9456, 00	2.6867, 00			
	2	0.0000, 00	0.0000, 00			
14	1	5.7411, 00	0.0000, 00			
	2	0.0000, 00	4.0000, 00			

VECTOR DENOMINATOR COEFFICIENT MATRIX

NODE					
0	0.9499, 00	9.0000, 00	2.2999, 01	1.8000, 01	4.0000, 00
1	0.9999, 00	7.4305, 00	1.4250, 01	6.1388, 00	
2	1.5694, 00	8.7499, 00	1.1851, 01	4.0000, 00	
3	0.9999, 00	5.2666, 00	5.0000, 00		
4	2.1638, 00	9.2500, 00	6.1388, 00		
5	1.5694, 00	5.9043, 00	3.1931, 00		
6	2.9456, 00	8.6679, 00	4.0000, 00		
7	0.9999, 00	3.5218, 00			
8	1.6448, 00	5.0000, 00			
9	2.1638, 00	4.9343, 00			
10	4.3156, 00	6.1388, 00			
11	1.5694, 00	3.4699, 00			
12	2.3344, 00	3.1931, 00			
13	2.9456, 00	2.6867, 00			
14	5.9811, 00	4.0000, 00			

DRIVING POINT ADMITTANCE DENOMINATOR COEFFICIENT MATRIX

NODE				
0	4.0000, 00	2.7000, 01	4.6000, 01	1.8000, 01
1	1.0678, 00	5.2126, 00	4.2171, 00	
2	4.6991, 00	1.5769, 01	7.8053, 00	
3	2.2443,-01	7.7055,-01		
4	6.4782,-01	1.1876, 00		
5	8.6280,-01	1.5818, 00		
6	1.9419, 00	1.4241, 00		
7	1.1681,-02			
8	8.8312,-02			
9	1.2698,-01			
10	2.6017,-01			
11	1.4736,-01			
12	4.0160,-01			
13	3.9065,-01			
14	1.2539,-01			

END OF RUN - - DATE: 02 APR 65 ELAPSED TIME 30 SECONDS

As an experimental verification of the computer program a double notch filter was designed using the normalized transfer function

$$U = \frac{(s^2 + 1)(s^2 + 4)}{(s^2 + 0.04s + 1)(s^2 + 0.08s + 4)} ,$$

which produces a notch at j and at 2j.

Following Hazony and Joseph[7] this can be re-written as

$$U = \frac{B\ t_{32}}{B(1-A\ t_{31})} .$$

Choosing $A = 1$ and $B = s^4 + 9s^3 + 23s^2 + 18s + 4$ an RC transfer function vector is obtained which is almost identical to that of Example 2.2:

$$\underline{t} = \left[\frac{8.88s^3+17.9968s^2+17.76s}{s^4+9s^3+23s^2+18s+4} , \frac{s^4+5s^2+4}{s^4+9s^3+23s^2+18s+4} \right]$$

The driving point impedance was chosen to be

$$Y_{33} = \frac{s^4 + 9s^3 + 23s^2 + 18s + 4}{4s^3 + 27s^2 + 46s + 18} .$$

This network was synthesized by the computer and in a subprogram the notch frequencies were shifted to 60 and 120 cps and the component values were magnitude scaled by a factor of 1.5×10^{-6}.

In constructing the network the resistors and capacitors were chosen with a 1 % tolerance and the unity gain amplifier used was a D.C. emitter follower circuit with A = 0.998, Z_{in} = 30 Meg, and Z_{out} = 1 k. The response of this circuit is shown in Figure A.2—— some 60 cps noise hampered the response at that frequency.

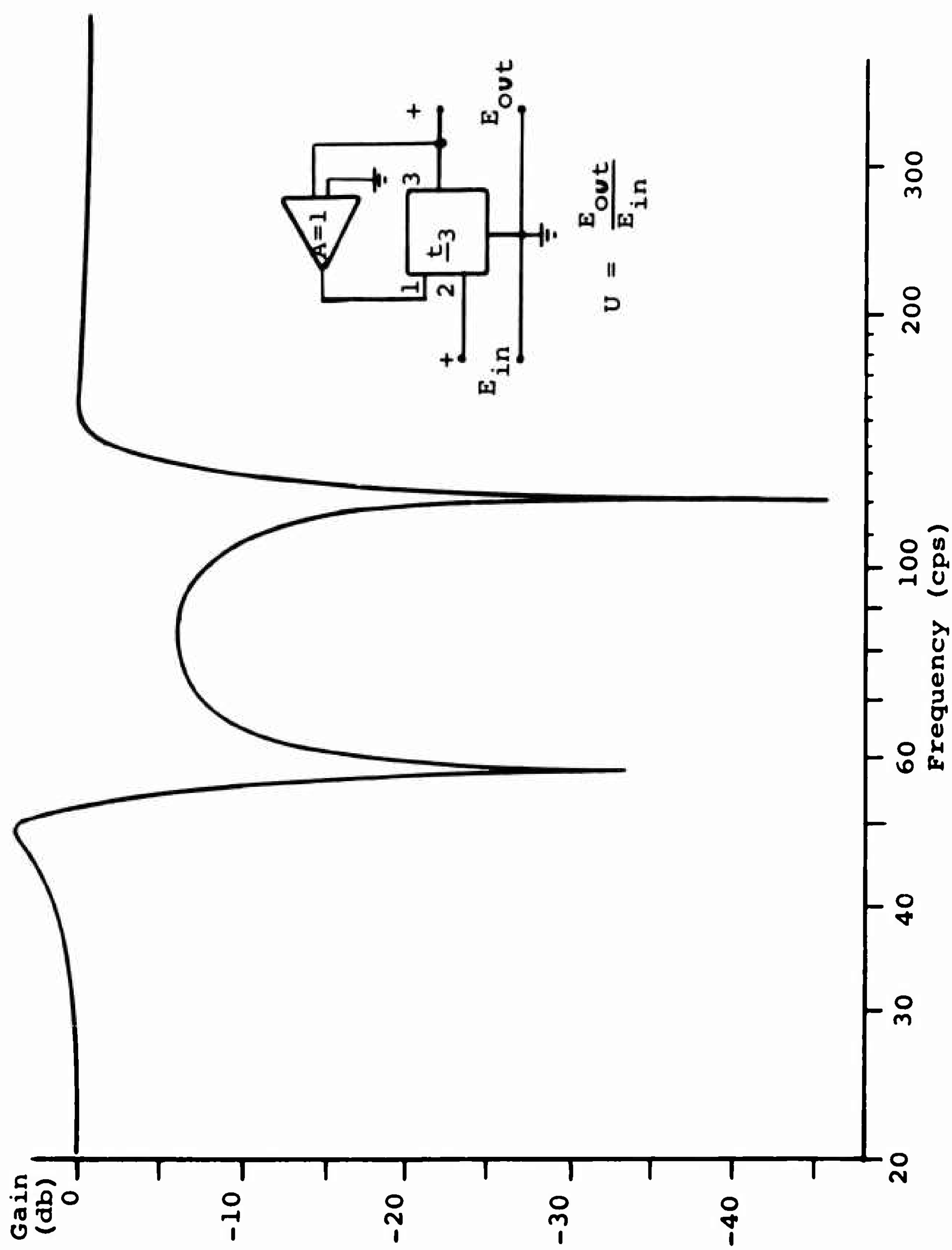


Fig. A.2 Experimental transfer response of the double notch filter.

```
BLOCK 2   LEVEL 2
M=N INPUTS$
TOP=(2**M)-2$
DONE=(2**(M-1))-2$
BEGIN
REAL ARRAY A(0..R,0..TOP,0..M), B,D(0..R,0..TOP),CAP,RES(0..TOP),
ZY,ZC(DONE..TOP,0..M)$                                                  51
                                                    COMMENT
READ DATA AND PRINT IT
                                                    $
READ(FOR J=(1,1,M) DO FOR K=(0,1,R) DO A(K,0,J),
      FOR J=(0,1,R) DO B(J,0),
      FOR J=(0,1,R-1) DO D(J,0))$
WRITE(FRMT1,FOR K=(0,1,R) DO B(K,0))$
FOR J=(1,1,M) DO WRITE(FRMT2,J,FOR K=(0,1,R) DO A(K,0,J))$
WRITE(FRMT3,FOR K=(0,1,R) DO B(K,0))$
WRITE(FRMT4,FOR K=(0,1,R-1) DO D(K,0))$
                                                    COMMENT
START THE SYNTHESIS
                                                    $
FOR U=(0,U+2,DONE) DO BEGIN FOR P=(U/2,1,U) DO BEGIN                    52
FOR I=(0,1,R) DO FOR J=(1,1,M) DO A(I,P,U)=A(I,P,0)+A(I,P,J)$           53
                                                    COMMENT
CALCULATE POSSIBLE OPTIMIZING LAMBDA AND CHOOSE CLOSSEST ONE TO 1/2
                                                    $
FOR X=(1,1,R-1) DO FOR Y=(0,1,M) DO BEGIN
BESTLAM=0 $                                                             84
DENOM=B(X,P)-B(0,P)*D(X,P)/D(0,P)-B(R,P)*D(X-1,P)/D(R-1,P)$
NUMER1=A(X,P,Y)-B(0,P)*D(X,P)/D(0,P)$
NUMER2=A(X,P,Y)+B(0,P)*D(X,P)/D(0,P)-B(X,P)$
IF DENOM EQL 0 THEN GO TO JUMPOUT $
LAM=NUMER1/DENOM $
DEL=-NUMER2/DENOM $
IF (LAM GEQ 1) OR (LAM LEQ 0) THEN LAM =0 $
IF (DEL GEQ 1) OR (DEL LEQ 0) THEN DEL =0 $
IF ABS(DEL-0.5) LSS ABS(LAM-0.5) THEN LAM=DEL $
IF ABS(LAM-0.5) LSS ABS(BESTLAM-0.5) THEN BESTLAM=LAM $
JUMPOUT..
END PICK OF BEST LAMBDA $                                               E4   C
IF BESTLAM EQL 0 THEN BESTLAM =0.5 $
                                                    COMMENT
REMOVE RC COMPONENTS AND CALCULATE NEW ADMITTANCES
                                                    $
LAM=BESTLAM $
DEL=1-BESTLAM $
B(0,2*P+1)=B(0,P)$
FOR I=(1,1,R-1) DO B(I,2*P+1)=B(I,P)*LAM
   +(B(0,P)*D(I,P)*DEL/D(0,P)) - (B(R,P)*D(I-1,P)*LAM/D(R-1,P))$
CAP(2*P+1)=B(R-1,2*P+1)/D(R-1,P)$
FOR I=(0,1,R-2) DO D(I,2*P+1)=D(I,P)-B(I,2*P+1)/CAP(2*P+1) $
FOR I=(0,1,R-2) DO B(I,2*P+2)=B(I+1,P)*DEL
   + B(R,P)*D(I,P)*LAM/D(R-1,P) - B(0,P)*D(I+1,P)*DEL/D(0,P)$
B(R-1,2*P+2)=B(R,P)$
RES(2*P+2)=D(0,P)/B(0,2*P+2)$
```

```
MONITOR SYSTEM -- 05KB4 03/29/65
    RUN     D HILBERMAN                                                              TIME:     12:59:04   DATE:   02 APR 65

  ALG      FILTER
      ALGOL        MARCH 21,1965          INTERFACE   FEBRUARY 15,1965      PASS2   DECEMBER 23, 1964
                                                                       COMMENT
             THIS ALGOL 60 PROGRAM WILL PRODUCE A FIALKOW-GERST SYNTHESIS RE
             FIALKOW ET AL, IEEE TRANS CKT THEORY, VOL CT-11 NO 1 MARCH 1964,
             FROM A GIVEN RC TRANSFER FUNCTION VECTOR AND ITS DRIVING POINT
             ADMITTANCE.   THE RESISTORS REMOVED ARE NUMBERED SEQUENTIALLY.
             THUS NODE 0 IS 0, NODE 1 IS 1, 2 IS 2, 11 IS 3, 12 IS 4, 21 IS 5,
             22 IS 6, 111 IS NUMBERED 7, ETC.  THE SUBSCRIPT 0 REFERS TO GROUND.
                                                                       $
                                                                               COMMENT  $
             REAL RET,SUM,MXA,MNB,MXB,DIFF,BESTLAMS
BLOCK 1    LEVEL 1                                                                          80
             REAL LAM,DEL,SUMY,SUMC,DENOM,NUMER1,NUMER2S
             INTEGER U,P,V,R,M,I,J,K,L,X,Y,TOP,DONE, DEGREE,NUMINPUTS,MNP,MXP,MXKS
             FORMAT FRMT0('DEGREE=',I3,X5,'NO. INPUTS=',I3,A1.0)S
             FORMAT FRMT1( 'THE DENOMINATOR COEFFICIENTS ARE',A2.0,                                 0
                 (5R14.6,A1.0))S
             FORMAT FRMT2('THE NUMERATOR COEFFICIENTS OF ENTRY',I3,' ARE',A2.0,                     0
                 (5R14.8,A1.0))S
             FORMAT FRMT3('THE NUMERATOR COEFFICIENTS OF Y(0,0) ARE',A2.0,                          0
                 (5R14.8,A1.0))S
             FORMAT FRMT4('THE DENOMINATOR COEFFICIENTS OF Y(0,0) ARE',A2.0,                        0
                 (5R14.8,A1.0))S
             FORMAT FRMT5(I3,X2,I3,X8,'NONE',X13,'NONE',A1.0)S
             FORMAT FRMT6(I3,X2,I3,X3,R14.8,X8,'NONE',A1.0)S
             FORMAT FRMT7(I3,X2,I3,X8,'NONE',X8,R14.8,A1.0)S
             FORMAT FRMT8(I3,X2,I3,X3,R14.8,X3,R14.8,A1.0)S
             FORMAT FRMT9('THE FOLLOWING COMPONENTS ARE IN FARADS AND OHMS',A2.0,                   0
                 'THE FREQUENCY IS ONE RADIAN/SECOND',A1.0,                                         0
                 'THE TREE COMPONENTS ARE',A1.0,                                                    0
                 'NODE',X8,'CAP',X13,'NODE',X8,'RES',A1.0)S
             FORMAT FRMT10(I3,X3,R14.8,X8,I3,X3,R14.8,A1.0)S
             FORMAT FRMT11('THE TERMINATING COMPONENTS ARE',A2.0,                                   0
                 'NODE INPUT',X6,'CAP',X14,'RES',A1.0)S
             FORMAT FRMT12(E)S
             FORMAT FRMT13('VECTOR NUMERATOR COEFFICIENT MATRIX',A2.0,                              0
                 'NODE INPUT',A1.0)S
             FORMAT FRMT14(I3,X3,I3,X4,(9R12.5,A1.0))S
             FORMAT FRMT15(X6,I3,X4,(9R12.5,A1.0))S
             FORMAT FRMT16('VECTOR DENOMINATOR COEFFICIENT MATRIX',A2.0,                            0
                 X6,'NODE',A1.0)S
             FORMAT FRMT17('DRIVING POINT ADMITTANCE DENOMINATOR COEFFICIENT MATRIX',               0
                 A2.0,X6,'NODE',A1.0)S
                                                               COMMENT $
             START..READ(DEGREE,NUMINPUTS)S
             WRITE(FRMT12)S
             M=DEGREES
```

```
       END P $                                                                    E3   C
       R=R-1 $
       END U. REMOVAL OF RC TREE $                                                E2   C
       FINALSYNTHESIS..
       WRITE(FRMT9)$
       FOR P=(0,1,DONE) DO WRITE(FRMT10,2*P+1,CAP(2*P+1),2*P+2,RES(2*P+2))$
                                                         COMMENT
       FINISH SYNTHESIS NOW THAT TRANSFER FUNCTIONS ARE OF UNITY DEGREE
                                                         $
       WRITE(FRMT11)$
       FOR U=(DONE+1,1,TOP) DO BEGIN
       SUMY=0$                                                                    B23
       SUMC=0$
       FOR J=(1,1,M) DO SUMY=SUMY + A(1,U,J)$
       FOR J=(1,1,M) DO SUMC=SUMC + A(0,U,J)$
       ZY(U,0)=(B(1,U)-SUMY)/D(U,U) $
       ZC(U,0)=(B(0,U)-SUMC)/D(U,U) $
       FOR J=(1,1,M) DO ZY(U,J)=A(1,U,J)/D(0,U) $
       FOR J=(1,1,M) DO ZC(U,J)=A(0,U,J)/D(0,U) $
       FOR J=(0,1,M) DO BEGIN
            IF ZY(U,J) EQL 0 THEN BEGIN                                           B24
                 IF ZC(U,J) EQL 0 THEN BEGIN                                      B25
                      WRITE(FRMT5,U,J)$ GO TO NEXT $                              B26
                                       END $                                      E26
                 WRITE(FRMT6,U,J,ZC(U,J))$ GO TO NEXT
                                       ENDS                                       E25
            IF ZC(U,J) EQL 0 THEN BEGIN
                 WRITE(FRMT7,U,J, 1/ZY(U,J))$                                     B27
                 GO TO NEXT         ENDS                                          E27
            WRITE(FRMT8,U,J,ZC(U,J), 1/ZY(U,J))$
       NEXT..            END $                                                    E24
       END TRANSFER SYNTHESIS $                                                   E23  C
       WRITE(FRMT12)$
       COMMENT NOW PRINT OUT PRINTED CIRCUIT BOARDS
       BEGIN
       REAL PI,SCALE,FREQS
BLOCK 3    LEVEL 3                                                                B28
       INTEGER CARDS,CDNUM,GS
       INTEGER ARRAY TERM(1..14)$
       FORMAT FMT0(X04,'TERM 8 IS THE OUTPUT',A1.0)$
       FORMAT FMT1(X04,'TIE TERM 8 TO TERM',I3,' OF CARD',I3,A1.0)$
       FORMAT FMT2(X6,B4('-'),A1.0,                                               Q
       X6,'II',X9,'.',X71,'I',A1.0,                                               Q
       X6,'II',7('.'),' . . ',7('.'), 60('-'),' I',A1.0,                          Q
       X6,'II',X7,'.    .',X7,'II',X57,'I  I', A1.0,                              Q
       X9,'R=',T14.5,'  II',X15,'.', X15,'II',X24,'I  I', A1.0 ,                  Q
       X6,'II',X19,'I',13('.'), ' . . ',13('.'),'II',X24,'I  I',A1.0,             Q
       X6,'II C=',T14.5,'  II',X13,'.    .',X13,'II',X24,'I  I',X4,               Q
                   'TOP VIEW OF TREE CARD NO. ',I4,A1.0,                          Q
       X6,'II',X8,'..',X9,'II',X11, 'R=',T14.5,X4,'II',X24,'I  I',A1.0,           Q
       X6,'II',19('.'),'II',X31,'II',X24,'I  I',X4,'FREQUENCY =',R14.8,           Q
                   ' CPS',A1.0,                                                   Q
       X6,'II',X8,'..',X9,'II',X31,'II',X24,'I  I',A1.0,                          Q
       X6,'II',X52,'II',X24,'I  I',A1.0,                                          Q
```

```
FOR I=(0,1,R-2) DO D(I,2*P+2)=D(I+1,P)-RES(2*P+2)*B(I+1,2*P+2) $
                                                     COMMENT
CALCULATE NEW TRANSFER FUNCTION NUMERATOR
                                                       $
FOR I=(1,1,R-1) DO BEGIN
MXA=0 $                                                                          B5
FOR K=(1,1,M) DO MXA=MAX(MXA,A(I,P,K))$
FOR K=(1,1,M) DO IF A(I,P,K) EQL MXA THEN MXK=K $
MXB=MAX(B(I,2*P+1),B(I-1,2*P+2))$
MNB=MIN(B(I,2*P+1),B(I-1,2*P+2))$
IF MXB EQL B(I-1,2*P+2) THEN MXP=2*P+2 ELSE MXP = 2*P+1 $
IF MNB EQL B(I-1,2*P+2) THEN MNP=2*P+2 ELSE MNP = 2*P+1 $
SUM=A(I,P,0) $
IF SUM LEQ MNB THEN BEGIN
     FOR K=(1,1,M) DO BEGIN A(I,MNP,K)=A(I,P,K)$ A(I,MXP,K)=0 $ ENDS      B6   B7   E7
     GO TO XFERDONE $        ENDS                                        E6
IF SUM LEQ MXB THEN BEGIN
     FOR K=(1,1,M) DO BEGIN A(I,MXP,K)=A(I,P,K)$ A(I,MNP,K)=0 $ ENDS      B8   B9   E9
     GO TO XFERDONE $        ENDS                                        E8
FOR J=(1,1,M) DO BEGIN
     IF A(I,P,J) EQL MNB THEN BEGIN                                      B10
     FOR K=(1,1,M) DO BEGIN A(I,MXP,K)=A(I,P,K)$ A(I,MNP,K)=0 $ ENDS      B11  B12  E12
     A(I,MNP,J)=MNB $
     A(I,MXP,J)=0 $
     GO TO XFERDONE $        ENDS                                        E11
     IF A(I,P,J) EQL MXB THEN BEGIN
     FOR K=(1,1,M) DO BEGIN A(I,MNP,K)=A(I,P,K)$ A(I,MXP,K)=0 $ ENDS      B13  B14  E14
     A(I,MXP,J)=MXB $
     A(I,MNP,J)=0 $
     GO TO XFERDONE $        ENDS                                        E13
                   ENDS                                                  E10
IF MXA GTR MNB THEN BEGIN
     FOR K=(1,1,M) DO BEGIN A(I,MXP,K)=A(I,P,K)$ A(I,MNP,K)=0 $ ENDS      B15  B16  E16
     A(I,MNP,MXK) = MNB $
     A(I,MXP,MXK)=MXA-MNB $
     GO TO XFERDONES         ENDS                                        E15
IF (SUM GTR MXB) AND (MXA LSS MNB) THEN BEGIN
     SUM=0$                                                              B17
     FOR K=(1,1,M) DO BEGIN
         IF (SUM + A(I,P,K)) GEQ MNB THEN BEGIN                          B18
         DIFF=MNB - SUM $                                                B19
         FOR J=(1,1,K-1) DO BEGIN A(I,MNP,J)=A(I,P,J)$A(I,MXP,J)=0$ENDS   B20  E20
         A(I,MNP,K) = DIFF $
         A(I,MXP,K)=A(I,P,K) - DIFF $
         FOR J=(K+1,1,M) DO BEGIN A(I,MXP,J)=A(I,P,J)$A(I,MNP,J)=0$ENDS   B21  E21
         GO TO XFERDONE $                      ENDS                      E19
         SUM = SUM + A(I,P,K) $
                                        END $                            E18
                                        END $                            E17
XFERDONE..
END TRANSFER FUNCTION CALCULATION $                                      E5   C
FOR K=(1,1,M) DO FOR I=(1,1,R) DO A(I-1,2*P+2,K)=A(I,2*P+2,K) $
FOR K=(1,1,M) DO BEGIN A(0,2*P+1,K)=A(0,P,K)$                           B22
      A(R-1,2*P+2,K)=A(R,P,K)$ ENDS                                      E22
```

```
X6,'II',X4,'I',7('.'),' . . ',8('.'),'II',X48,'I',A1.0,
X6,'II',X4,'I',X7,'.    .',X8,'II',X48,'I',X12,'TERM  8 TO GROUND',A1.0,
X6,'II',X4,'I',X2,'R=',T14.5,X2,'II',X48,'I',A1.0,
X6,'II',X4,'I',X20,'I',45('-'),X4,'I',X12,'TERM  1 TO INPUT 1',A1.0,
X6,'II',X4,'II',X9,'..',X9,'I',44('-'),'I',X4,'I',A1.0,
X6,'II',X4,'II',20('.'),'II',X7,'.',X24,'.',X10,'I',X4,'I',X12,
            'TERM 15 TO INPUT 2',A1.0,
X6,'II',X4,'I',X9,'..',X9,'II',X7,'.',X24,'.',X10,'I',X4,'I',A1.0,
X6,'II',X4,'I',X2,'C=',T14.5,X2,'II',X7,'.',X24,'.',X10,'I',X4,17('-'),
            '  15',A1.0,
X6,'II',X4,'I',X20,'II',X7,'.',X24,'.',X10,'I',A1.0,
X6,'II',X4,'I',X20,'II',X7,'.',X24,'.',X10,'I',X16,5('-'),'  14',A1.0,
X6,'II',X4,'I',X20,'II',X7,'.',X24,'.',X10,'I',A1.0,
X6,'II',X4,'II',X9,'.',X10,'II',X7,'.',X24,'.',X10,22('-'),'  13',A1.0,
X6,'II',4('.'),'II',7('.'),' . . ',8('.'),'II',X5,5('.'),X20,'.',A1.0,
X6,'II',X4,'II',X7,'.    .',X8,'II',X5,5('.'),X22,'.',X27,5('-'),'  12',
            A1.0,
X6,'II',X4,'I',X2,'R=',T14.5,X2,'II',X7,'. C=',T14.5,X9,'. R=',T14.5,
            A1.0)$
FORMAT F=T38(X6,'II',X4,'II',X20,'II',X7,'.',X24,'.',X27,5('-'),
            '  11',A1.0,
X6,'II',X4,'I',X9,'..',X9,'II',X7,'.',X24,'.',A1.0,
X6,'II','....','II',20('.'),'II', X7,'.',X24,'.',X27,5('-'),'  10',A1.0,
X6,'II',X4,'I',X9,'..',X9,'II',X7,'.',X24,'.',A1.0,
X6,'II',X4,'II',X2,'C=',T14.5,X2,'II',X7,'.',X24,'.',X27,5('-'),'   9',
            A1.0,
X6,'II',X4,'II',X20,40('-'),A1.0,
X6,'II',X4,'II',X20,61('-'),'   8',A1.0,
X6,'II',X4,'I',X20,40('-'),A1.0,
X6,'II',X4,'I',X20,'II',X7,'.',X24,'.',X27,5('-'),'   7',A1.0,
X6,'II',X4,'II',X9,'..',X9,'II',X7,'.',X24,'.',A1.0,
X6,'II','....','II',20('.'),'II', X7,'.',X24,'.',X27,5('-'),'   6',A1.0,
X6,'II',X4,'II',X9,'..',X9,'II',X7,'.',X24,'.',A1.0,
X6,'II',X4,'I',X2,'C=',T14.5,X2,'II',X7,'.',X24,'.',X27,5('-'),'   5',
            A1.0,
X6,'II',X4,'I',X20,'II',X7,'. C=',T14.5,X7,'.',X3,'R=',T14.5,A1.0,
X6,'II',X4,'I',X9,'.',X10,'II',X7,'.',X22,'.',X29,5('-'),'   4',A1.0,
X6,'II',4('.'),'II',7('.'),' . . ',8('.'),'II',X5, 5('.'),X22,'.',A1.0,
X6,'II',X4,'II',X7,'.    .',X8,'II',X5,5('.'),X24,'.',X8,22('-'),'   3',
            A1.0,
X6,'II',X4,'I',X2,'R=',T14.5,X2,'II',X7,'.',X24,'.',X10,'I',A1.0,
X12,'II',X20,'II',X7,'.',X24,'.',X10,'I',X16,5('-'),'   2',A1.0,
X12,'II',X20,'II',X7,'.',X24,'.',X10,'I',A1.0)$
FORMAT F=T39(X12,'II',X20,'II',X7,'.',X24,'.',X10,'I',X4,17('-'),
            '   1',A1.0,
X12,'II',X20,'II',X7,'.',X24,'.',X10,'I',X4,'I',A1.0,
X12,'II',X9,'..',X9,'II',X7,'.',X24,'.',X10,'I',X4,'I',A1.0,
X12,'II',20('.'),   'II',X7,'.',X24,'.',X10,'I',X4,'I',A1.0,
X12,'II',X9,'..',X9,'I',44('-'),'I',X4,'I',X4,'FREQUENCY =',R14.8,
            ' CPS',A1.0,
X12,'II',X2,'C=',T14.5,X2,'I',45('-'),X4,'I',A1.0,
X12,'II',X20,'II',X48,'I',X4,'SCALE FACTOR =',R14.8,A1.0,
X12,'II',X9,'.',X10,'II',X48,'I',A1.0,
X12,'II',7('.'),' . . ',8('.'),'II',X48,'I',X4,'RESISTANCE IN KILOHMS',
```

```
X6,30('-'),X24,'II',X24,'I  I',A1.0,
X35,'I     C=',T14.5,X4,'II',X24,'I',X2,16('-'),'  15',A1.0,
X6,26('-'),X3,'I',X24,'II',X6,'I.',X16,'I',A1.0,
X6,'II',X9,'.',X13,'I   I',X8,'..',X14,'II',X6,'I.',X16,19('-'),'  14',
          A1.0,
X6,'II',7('.'),' . . ',7('.'),'II..I...I',24('.'),'II',X6,'I.',A1.0,
X6,'II',X7,'.    .',X7,'II  I   I',X8,'..',X14,45('-'),'  13',A1.0,
X9,'R=',T14.5,'  II  I   I',X33,'.',A1.0,
X6,'II',X19,'II  I  ',71('-'),'  12',A1.0,
X6,'II C=',T14.5,'  II  I',X37,'.',A1.0,
X6,'II',X8,'..',X9,'II  ',74('-'),'  II', A1.0,
X6,'II',19('.'),'II',X40,'R=',T14.5,A1.0,
X6,'II',X8,'..',X9,78('-'),'  10',A1.0,
X6,'II',X61,'.',A1.0) $
FORMAT FMT2A(X6,99('-'),'   9',A1.0,
X69,'.',A1.0,
X65,'--+-+',35('-'),'   8',A1.0,
X67,'.',A1.0,
X6,99('-'),'   7',A1.0,
X6,'II',X9,'.',X49,'.',A1.0,
X6,'II',7('.'),' . . ',7('.'),78('-'),'   6',A1.0,
X6,'II',X7,'.    .',X7,'II',X34,'---+---',A1.0,
X9,'R=',T14.5,'  II  ',74('-'),'   5',A1.0,
X6,'II',X19,'II  I',X33,'---+--   C=',T14.5,A1.0,
X6,'II C=',T14.5,'  II  I   ',70('-'),'   4',A1.0,
X6,'II',X8,'..',X9,'II  I   I',X31,'.',A1.0,
X6,'II',19('.'),'II  I   I',X8,'.',X15,45('-'),'   3',A1.0,
X6,'II',X8,'..',X9,'II..I...I',6('.'),' . . ',13('.'),'II',
          X5,'.',X10,'I',A1.0,
X6,'II',X23,'I   I',X6,'.   .',X13,'II',X5,'.',X10,'I',X7,19('-'),
          '   2',A1.0,
X6,26('-'),X3,'I',X24,'II',17('-'),X7,'I',A1.0,
X35,'I     R=',T14.5,X4,'II',X24,'I  ',16('-'),'   1',A1.0,
X6,30('-'),X24,'II',X24,'I  I',A1.0,
X6,'II',X9,'.',X42,'II',X24,'I  I',A1.0,
X6,'II',7('.'),' . . ',7('.'),'II',X31,'II',X24,'I  I',A1.0,
X6,'II',X7,'.    .',X7,'II',X11,'C=',T14.5,X4,'II',X24,'I  I',X4,
          'RESISTANCE IN KILOHMS',A1.0,
X9,'R=',T14.5,'  II',X31,'II',X24,'I  I',X4,'CAPACITANCE IN MICROFARADS'
          ,A1.0)$
FORMAT FMT2B(Y6,'II',X19,'II',X15,'..',X14,'II',X24,'I  I',X4,A1.0,
X6,'II C=',T14.5,'  II',31('.'),'II',X24,'I  I',X4,'SCALE =',R14.8,
          A1.0,
X6,'II',X8,'..',X9,'II',X15,'..',X14,'II',X24,'I  I',A1.0,
X6,'II', 19('.'),'II',X57,'I  I',A1.0,
X6,'II',X8,'..',X9,60('-'),'  I',A1.0,
X6,'II',X81,'I',A1.0,
X6,84('-'),A1.0)$
FORMAT FMT3(X4,79('-'), X4,'TOP VIEW OF FINAL CARD NO.',I3,A1.0,
X6,'II',X76,'I',A1.0,
X6,'II',X76,'I',X4,'CONNECT TERM  3 TO TERM',I3,' OF CARD',I3,A1.0,
X6,'II',X4,'II',X20,'II',X48,'I',A1.0,
X6,'II',X4,'II',X9,'.',X10,'II',X48,'I',X12,'TERM 13 TO TERM',I3,
          ' OF CARD',I3,A1.0,
```

```
        WRITE(FMT3,CARDS+(J-DONE)/2,TERM(Q),CONUM,TERM(Q-1),CONUM,ZY(J-1,1),
             ZC(J-1,1),ZY(J-1,2),ZC(J-1,0),ZY(J-1,0))$
        WRITE(FMT3A,ZC(J-1,2),ZC(J,2),ZC(J,0), ZY(J,0),ZY(J,2))$
        WRITE(FMT3B,FREQ,ZC(J,1),SCALE,ZY(J,1))$
                                 ENDS                                          E36
        JUNK..IF TOP LEQ 14 THEN GO TO JAILS
        I=K=L=DONE=0 $
        R=R+3$
        J=CARDS $
        FOR J=(0,1,J-2) DO BEGIN
        IF B*K GEQ DONE THEN BEGIN                                             B37
             R=R-3 $                                                           B38
             K=0 $
             V=2**(R-4) $
             DONE=B*V -2 $
             TOP=16*V -2 $
             CARDS=-ENTIER(-TOP/14.0) $
                             ENDS                                              E38
        L=7 + MOD(4*V -1-K, 8) $
        CONUM=CARDS - V - (K//8) $
        Q=TOP - B*K $
        IF CONUM EQL 1 THEN L=(Q-14)/4 $
        WRITE(FMT1, TERM(L),CONUM)$
        WRITE(FMT2,RES(Q),CAP(Q-1),CARDS-K,RES(Q/2-1),FREQ,
             CAP(Q/2-2),RES(Q-2),CAP(Q-3),RES((Q-6)/4))$
        WRITE(FMT2A,RES(Q-4),CAP((Q-10)/4),CAP(Q-5),RES(Q/2-3),CAP(Q/2-4),
             RES(Q-6))$
        WRITE(FMT2B,CAP(Q-7),SCALE)$
        K=K+1$
                                ENDS                                           E37
        JAIL..K=TOP - 14*(CARDS-1) $
        FOR J=(K+1,1,14) DO RES(J)=CAP(J)= B-30 $
        I=1$
        WRITE(FMT0)$
        WRITE(FMT2,RES(14),CAP(13),I,RES(6),FREQ,CAP(5),RES(12),CAP(11),RES(2))$
        WRITE(FMT2A,RES(10),CAP(1),CAP(9),RES(4),CAP(3),RES(8))$
        WRITE(FMT2B,CAP(7),SCALE)$
        END PRINTED CIRCUIT CARDS $                                            E28    C
END BLOCK 3

                                                        COMMENT
        PRINT OUT COEFFICIENT MATRICES
                                                          $
        M=DEGREE  $
        WRITE(FRMT12)$
        WRITE(FRMT13)$
        FOR U=(0,U+2,TOP) DO BEGIN
             FOR P=(U/2,1,U) DO BEGIN                                          B39
                  K=1 $                                                        B40
                  WRITE(FRMT14,P,K,FOR I=(0,1,R) DO A(I,P,K))$
                  FOR K=(2,1,M) DO WRITE(FRMT15,K,FOR I=(0,1,R) DO A(I,P,K))$
                                 ENDS                                          E40
             R=R-1 $
                            ENDS                                               E39
```

```
                    A1.0,                                                        @
X12,'II',X7,',     ,',X8,'II',X48,'I',A1.0,                                      @
X12,'II',X2,'R=',T14.5,X2,'II',X48,'I',X4,'CAPACITANCE IN MICROFARADS',         @
                    A1.0,                                                        @
X12,'II',X70,'I',A1.0,                                                           @
X12,'II',X70,'I',A1.0,                                                           @
X12,'II',71('-'),A1.0)$
FORMAT FMT4('TOO MANT INPUTS FOR PRINTED CIRCUIT FINAL CARDS',A1.0,E0)$
FORMAT FMT5('SCALED COMPONENTS IN MICROFARADS AND KILOHMS',A1.0)$
FORMAT FMT6('NODE',X6,'CAP',X13,'NODE',X8,'RES',A1.0)$
FORMAT FMT7(I3,X3,R14.8,X8,I3,X3,R14.8,A1.0)$
FORMAT FMT8('NODE INPUT',X6,'CAP',X14,'RES',A1.0)$
FORMAT FMT9(I3,X2,I3,X8,'NONE',X13,'NONE',A1.0)$
FORMAT FMT10(I3,X2,I3,X3,R14.8,X8,'NONE',A1.0)$
FORMAT FMT11(I3,X2,I3,X8,'NONE',X8,R14.8,A1.0)$
FORMAT FMT12(I3,X2,I3,X3,R14.8,X3,R14.8,A1.0)$
FORMAT FMT13(E0)$
                                                   COMMENT $
I=1$FOR TERM(I)=3,13,2,6,10,14,1,4,5,7,9,11,12,15 DO I=I+1$
M=DEGREES
PI = 3.1415927 $
READ(SCALE,FREQ)$
FOR J=(DONE+1,1,TOP) DO FOR K=(0,1,M) DO BEGIN
     ZY(J,K)=SCALE*ZY(J,K)a3 $                                                   B29   I
     IF ZY(J,K) GTR 0 THEN ZY(J,K)=1/ZY(J,K) $
     ZC(J,K)=SCALE*ZC(J,K)a6 /(2*PI*FREQ) $                                      I
                                                   ENDS                          E29
FOR J=(1,1,TOP) DO BEGIN
     RES(J)=RES(J)/SCALEa3 $                                                     B30   I
     CAP(J)=SCALE*CAP(J)a6 /(2*PI*FREQ) $                                        I
                         ENDS                                                    E30
WRITE(FMT5)$
WRITE(FMT6)$
FOR J=(0,1,DONE) DO WRITE(FMT7,2*J+1,CAP(2*J+1),2*J+2,RES(2*J+2))$
WRITE(FMT8)$
FOR U=(DONE+1,1,TOP) DO FOR J=(0,1,M) DO BEGIN
     IF ZY(U,J) EQL 0 THEN BEGIN                                                 B31
          IF ZC(U,J) EQL 0 THEN BEGIN                                            B32
               WRITE(FMT9,U,J)$                                                  B33
               GO TO NEXTROW $  ENDS                                             E33
          WRITE(FMT10,U,J,ZC(U,J))$
          GO TO NEXTROW $   ENDS                                                 E32
     IF ZC(U,J) EQL 0 THEN BEGIN
          WRITE(FMT11,U,J,ZY(U,J))$                                              B34
          GO TO NEXTROW $   ENDS                                                 E34
     WRITE(FMT12,U,J,ZC(U,J),ZY(U,J))$
NEXTROW..                                          ENDS                          E31
WRITE(FMT13)$
CARDS=-ENTIER(-TOP/14.0) $
IF M GTR 2 THEN BEGIN WRITE(FMT4)$ GO TO JUNKS ENDS                              B35   E35
FOR J=(TOP,-2,DONE +1) DO BEGIN
     CONUM= CARDS - (TOP-J)//8$                                                  B36
     Q=7 + MOD(J-DONE-1, 8)$
     IF CONUM EQL 1 THEN U=J$
```

```
THE DENOMINATOR COEFFICIENTS ARE
 0.9999999, 00 9.0000000, 00 2.2999999, 01 1.8000000, 01 4.0000000, 00

THE NUMERATOR COEFFICIENTS OF ENTRY  1 ARE
 0.0000000, 00 8.8799999, 00 1.7996799, 01 1.7759999, 01 0.0000000, 00

THE NUMERATOR COEFFICIENTS OF ENTRY  2 ARE
 0.9999999, 00 0.0000000, 00 5.0000000, 00 0.0000000, 00 4.0000000, 00

THE NUMERATOR COEFFICIENTS OF Y(0,0) ARE
 0.9999999, 00 9.0000000, 00 2.2999999, 01 1.8000000, 01 4.0000000, 00

THE DENOMINATOR COEFFICIENTS OF Y(0,0) ARE
 4.0000000, 00 2.7000000, 01 4.6000000, 01 1.8000000, 01

THE FOLLOWING COMPONENTS ARE IN FARADS AND OHMS
THE FREQUENCY IS ONE RADIAN/SECOND
THE TREE COMPONENTS ARE
NODE        CAP              NODE        RES
  1     3.4104938,-01          2     2.5486725, 00
  3     1.1856221, 00          4     4.9349726,-01
  5     4.0910304,-01          6     1.5952959, 00
  7     4.7002804, 00          8     1.3644900,-01
  9     4.1546235, 00         10     1.5010854,-01
 11     2.1936721, 00         12     3.6960558,-01
 13     1.8865955, 00         14     3.2468323,-01

THE TERMINATING COMPONENTS ARE
NODE INPUT        CAP              RES
  7    0         NONE         9.7345446,-02
  7    1         NONE         3.3358022,-03
  7    2    8.5605738, 01         NONE
  8    0         NONE             NONE
  8    1    1.8624946, 01         NONE
  8    2         NONE         1.7662571,-02
  9    0         NONE         3.9679513, 01
  9    1    1.7041002, 01     2.5750999,-02
  9    2         NONE             NONE
 10    0         NONE             NONE
 10    1    1.6213961, 01     4.3358082,-02
 10    2         NONE             NONE
 11    0         NONE        -2.4724169, 06
 11    1    1.0649865, 01     4.2469179,-02
 11    2         NONE             NONE
 12    0         NONE             NONE
 12    1    5.8127440, 00     1.2576858,-01
 12    2         NONE             NONE
 13    0         NONE             NONE
 13    1    7.7382428, 00     1.4167890,-01
 13    2         NONE             NONE
 14    0    1.9139785, 00         NONE
 14    1    4.5785470, 01         NONE
 14    2         NONE         3.1348291,-02
```

```
     R=DEGREE $
     WRITE(FRMT12)$
     WRITE(FRMT16)$
     FOR U=(0,U+2,TOP) DO BEGIN
          FOR P=(U/2,1,U) DO WRITE(FRMT15,P,FOR I=(0,1,R) DO B(I,P))$       B41
          R=R-1 $
                         ENDS                                                E41
     R=DEGREE $
     WRITE(FRMT12)$
     WRITE(FRMT17)$
     FOR U=(0,U+2,TOP) DO BEGIN
          FOR P=(U/2,1,U) DO WRITE(FRMT15,P,FOR I=(0,1,R-1) DO D(I,P))$     B42
          R=R-1 $
                         ENDS                                                E42
     WRITE(FRMT12)$
     END WHOLE THING $                                                       E1    C
END BLOCK 2
                                                    COMMENT
     THIS COMPLETES THE SYNTHESIS OF ONE VECTOR OF THE MATRIX.
     GO BACK AND DO THE NEXT VECTOR.
                                                    $
     GO TO STRT $
     FINISH $                                                                F0
END BLOCK 1
***** THIS PROGRAM USED IMPLIED MULTIPLICATION! *****
COMPILATION COMPLETED
```

TOP VIEW OF FINAL CARD NO. 5

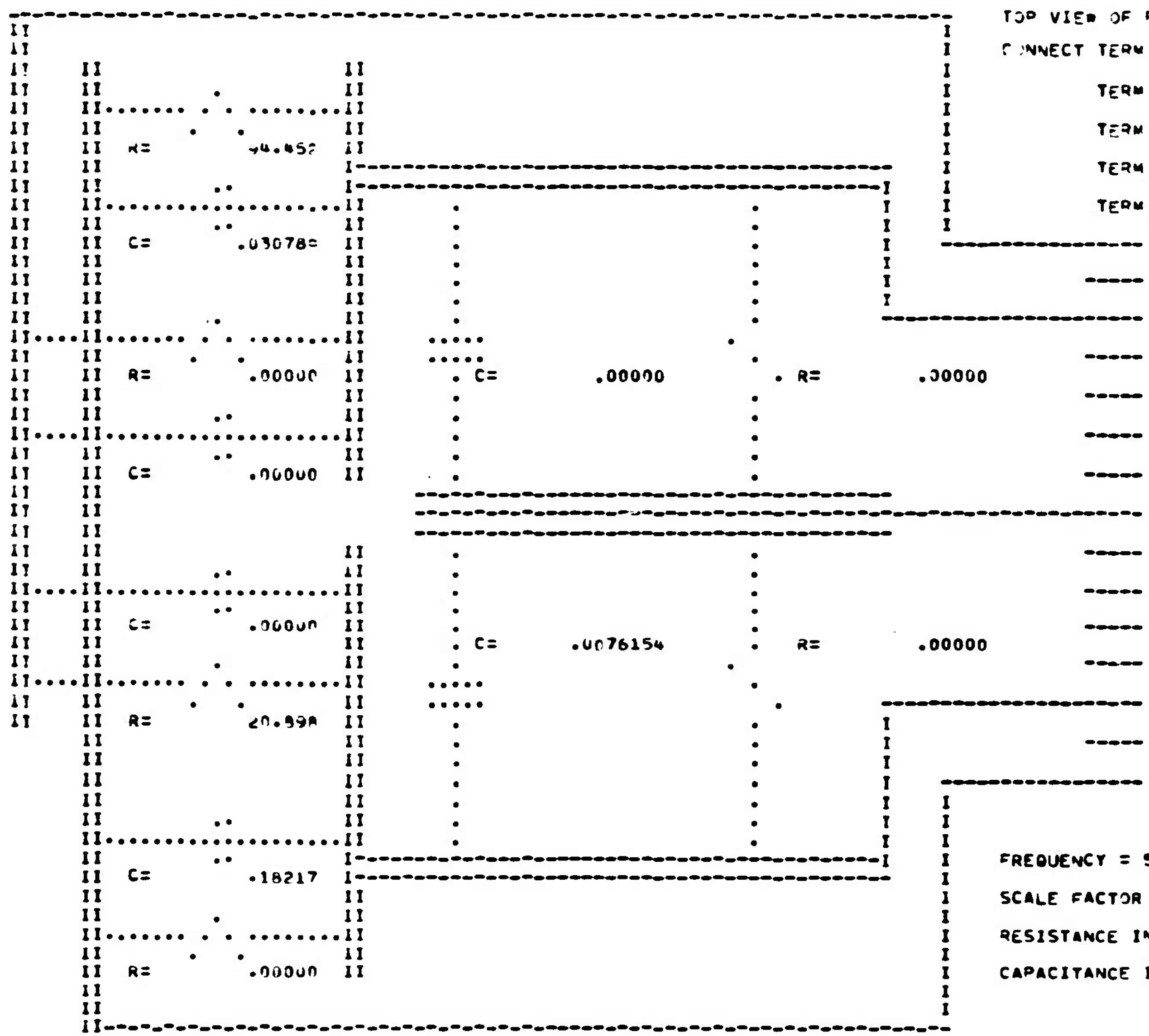
CONNECT TERM 3 TO TERM 15 OF CARD 1

TERM 13 TO TERM 12 OF CARD 1

TERM 8 TO GROUND

TERM 1 TO INPUT 1

TERM 15 TO INPUT 2



FREQUENCY = 5.9999999, 01 CPS

SCALE FACTOR = 1.4999999,-06

RESISTANCE IN KILOHMS

CAPACITANCE IN MICROFARADS

SCALED COMPONENTS IN MICROFARADS AND KILOHMS

NODE	CAP	NODE	RES
1	1.3569923, -03	2	1.6991150, 03
3	4.7174404, -03	4	3.2899817, 02
5	1.6277692, -03	6	1.0635306, 03
7	1.5701952, -02	8	9.0966004, 01
9	1.6530721, -02	10	1.0007235, 02
11	8.7283439, -03	12	2.4040372, 02
13	7.5065251, -03	14	2.1645549, 02

NODE	INPUT	CAP	RES
7	0	NONE	6.6896965, 01
7	1	NONE	2.2238681, 00
7	2	3.4061[illegible], -01	NONE
8	0	NONE	NONE
8	1	7.41[illegible], -01	NONE
8	2	NONE	1.1775047, 01
9	0	NONE	2.6453009, 04
9	1	6.7[illegible]93, -02	1.7167332, 01
9	2	NONE	NONE
10	0	NONE	NONE
10	1	6.[illegible], -02	2.8905388, 01
10	2	NONE	NONE
11	0	NONE	-6.0069376, -10
11	1	4.2374467, -02	2.8312787, 01
11	2	NONE	NONE
12	0	NONE	NONE
12	1	2.3128172, -02	8.3845722, 01
12	2	NONE	NONE
13	0	NONE	NONE
13	1	3.0789489, -02	9.4452608, 01
13	2	NONE	NONE
14	0	7.6154784, -03	NONE
14	1	1.8217459, -01	NONE
14	2	NONE	2.0898861, 01

TOP VIEW OF FINAL CARD NO. 3

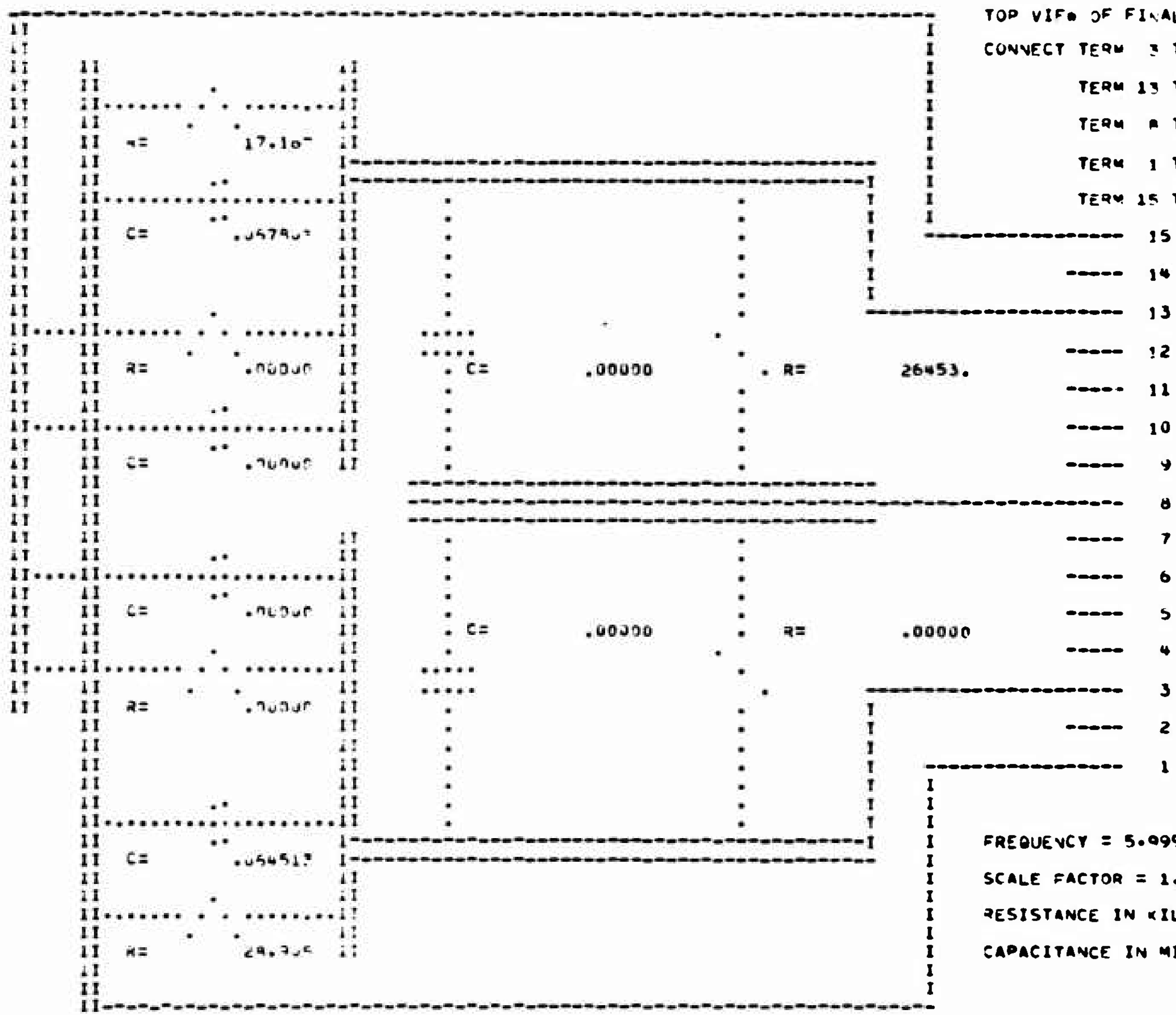
CONNECT TERM 3 TO TERM 7 OF CARD 1

TERM 13 TO TERM 5 OF CARD 1

TERM 8 TO GROUND

TERM 1 TO INPUT 1

TERM 15 TO INPUT 2

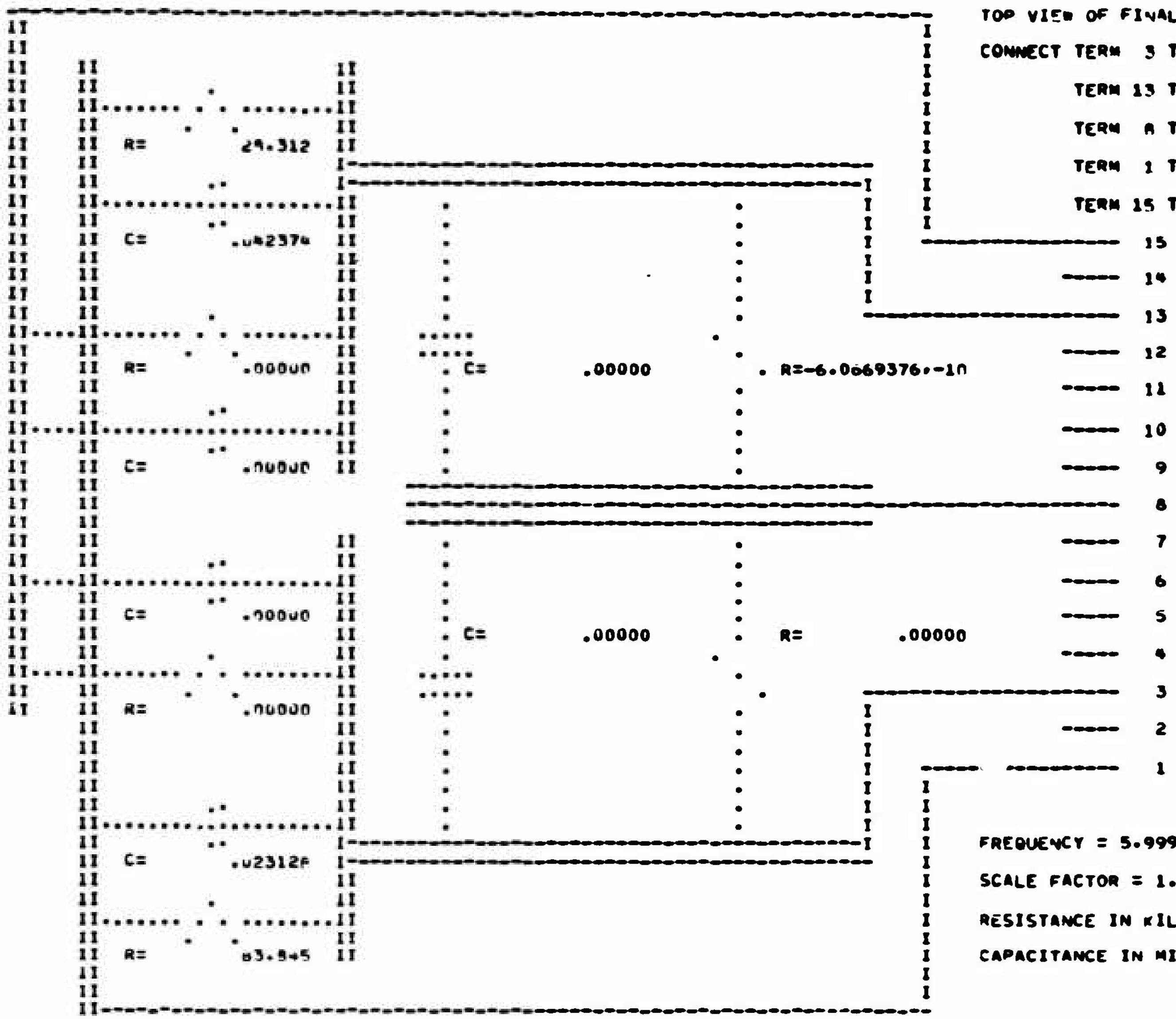


FREQUENCY = 5.9999999, 01 CPS

SCALE FACTOR = 1.4999999,-06

RESISTANCE IN KILOHMS

CAPACITANCE IN MICROFARADS



TOP VIEW OF FINAL CARD NO. 4

CONNECT TERM 3 TO TERM 11 OF CARD 1

TERM 13 TO TERM 9 OF CARD 1

TERM 8 TO GROUND

TERM 1 TO INPUT 1

TERM 15 TO INPUT 2

FREQUENCY = 5.9999999, 01 CPS

SCALE FACTOR = 1.4999999,-06

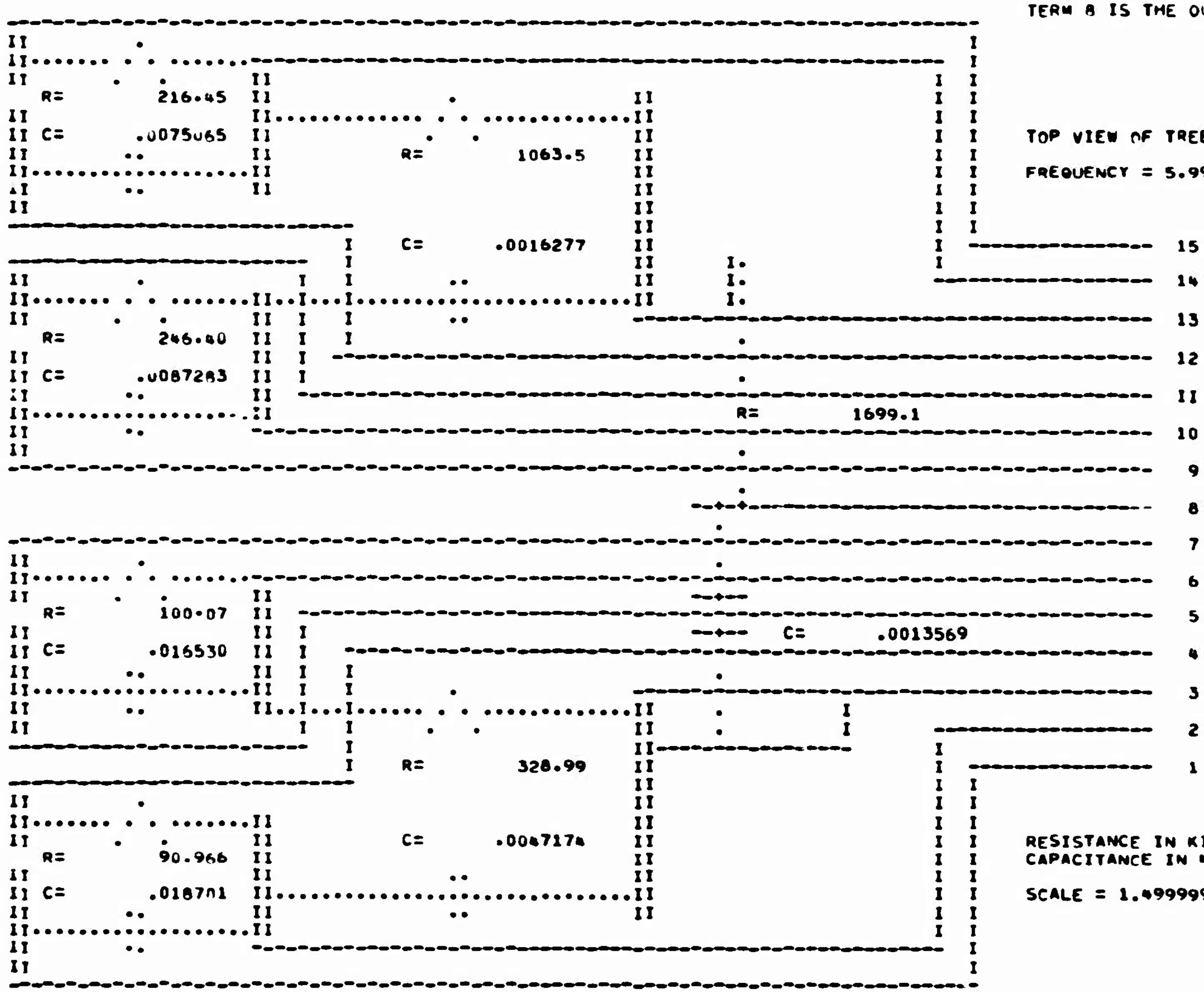
RESISTANCE IN KILOHMS

CAPACITANCE IN MICROFARADS

TERM 8 IS THE OUTPUT

TOP VIEW OF TREE CARD NO. 1

FREQUENCY = 5.9999999, 01 CPS



RESISTANCE IN KILOHMS
CAPACITANCE IN MICROFARADS

SCALE = 1.4999999,-06

TOP VIEW OF FINAL CARD NO. 2

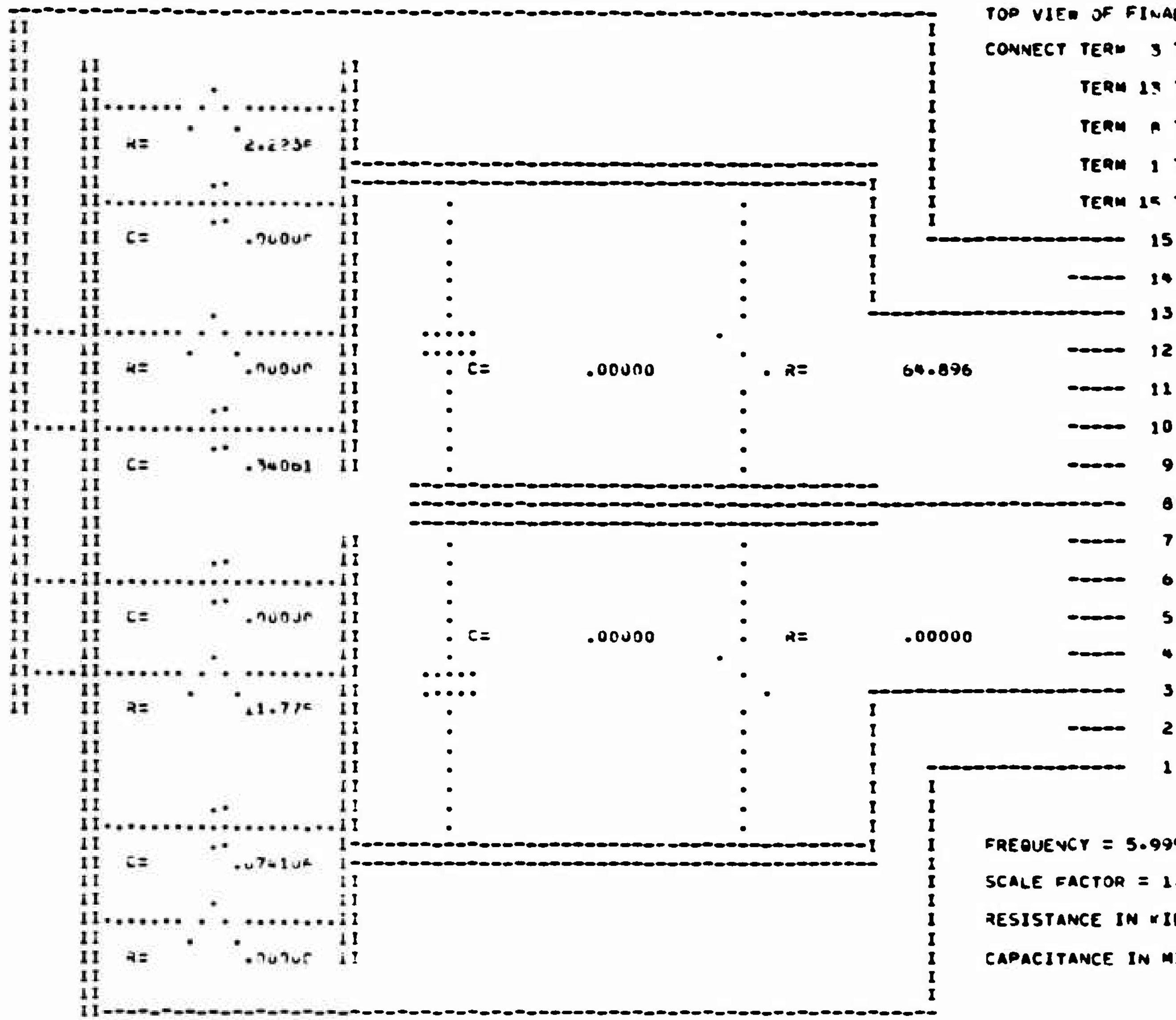
CONNECT TERM 3 TO TERM 4 OF CARD 1

TERM 13 TO TERM 1 OF CARD 1

TERM 8 TO GROUND

TERM 1 TO INPUT 1

TERM 15 TO INPUT 2



FREQUENCY = 5.9999999, 01 CPS

SCALE FACTOR = 1.4999999,-06

RESISTANCE IN KILOHMS

CAPACITANCE IN MICROFARADS

VECTOR DENOMINATOR COEFFICIENT MATRIX

NODE					
0	0.9999, 00	9.0000, 00	2.2999, 01	1.8000, 01	4.0000, 00
1	0.9999, 00	7.4305, 00	1.4250, 01	6.1388, 00	
2	1.5694, 00	9.7499, 00	1.1861, 01	4.0000, 00	
3	0.9999, 00	5.2666, 00	5.0000, 00		
4	2.1036, 00	9.2500, 00	6.1388, 00		
5	1.5694, 00	5.8043, 00	3.1931, 00		
6	2.9456, 00	8.6679, 00	4.0000, 00		
7	0.9999, 00	3.5216, 00			
8	1.6448, 00	5.0000, 00			
9	2.1036, 00	4.9343, 00			
10	4.3156, 00	6.1388, 00			
11	1.5694, 00	3.4659, 00			
12	2.3344, 00	3.1931, 00			
13	2.9456, 00	2.6867, 00			
14	5.9811, 00	4.0000, 00			

DRIVING POINT ADMITTANCE DENOMINATOR COEFFICIENT MATRIX

NODE				
0	4.0000, 00	2.7000, 01	4.6000, 01	1.8000, 01
1	1.0678, 00	5.2126, 00	4.2171, 00	
2	4.6491, 00	1.5769, 01	7.8053, 00	
3	2.2443,-01	7.7055,-01		
4	6.4782,-01	1.1876, 00		
5	8.5280,-01	1.5818, 00		
6	1.9419, 00	1.4241, 00		
7	1.1681,-02			
8	8.8312,-02			
9	1.2695,-01			
10	2.6617,-01			
11	1.4736,-01			
12	4.0160,-01			
13	3.8065,-01			
14	1.2539,-01			

END OF RUN - - PROJECT 3E026 PROGRAMMER 1 DATE: 02 APR 65 ELAPSED TIME 54 SECONDS

VECTOR NUMERATOR COEFFICIENT MATRIX

NODE	INPUT					
0	1	0.0000, 00	8.8799, 00	1.7996, 01	1.7759, 01	0.0000, 00
	2	0.9999, 00	0.0000, 00	5.0000, 00	0.0000, 00	4.0000, 00
1	1	0.0000, 00	7.3105, 00	9.2467, 00	6.1388, 00	
	2	0.9999, 00	0.0000, 00	5.0000, 00	0.0000, 00	
2	1	1.5094, 00	8.7409, 00	1.1621, 01	0.0000, 00	
	2	0.0000, 00	0.0000, 00	0.0000, 00	4.0000, 00	
3	1	0.0000, 00	5.1466, 00	0.0000, 00		
	2	0.9999, 00	0.0000, 00	5.0000, 00		
4	1	2.1638, 00	9.2467, 00	6.1388, 00		
	2	0.0000, 00	0.0000, 00	0.0000, 00		
5	1	1.5094, 00	5.4043, 00	3.1931, 00		
	2	0.0000, 00	0.0000, 00	0.0000, 00		
6	1	2.9456, 00	8.4279, 00	0.0000, 00		
	2	0.0000, 00	0.0000, 00	4.0000, 00		
7	1	0.0000, 00	3.5016, 00			
	2	0.9999, 00	0.0000, 00			
8	1	1.6448, 00	0.0000, 00			
	2	0.0000, 00	5.0000, 00			
9	1	2.1638, 00	4.9311, 00			
	2	0.0000, 00	0.0000, 00			
10	1	4.3156, 00	6.1388, 00			
	2	0.0000, 00	0.0000, 00			
11	1	1.5094, 00	3.4699, 00			
	2	0.0000, 00	0.0000, 00			
12	1	2.3344, 00	3.1931, 00			
	2	0.0000, 00	0.0000, 00			
13	1	2.9456, 00	2.6867, 00			
	2	0.0000, 00	0.0000, 00			
14	1	5.7411, 00	0.0000, 00			
	2	0.0000, 00	4.0000, 00			

REFERENCES

1. Fialkow, A., "Two Terminal Pair Networks Containing Two Kinds of Elements Only", Proc. Symp. Modern Network Synthesis, Polytechnic Inst. of Brooklyn, N.Y., pp. 50-65; 1952.

2. Fialkow, A., "Networks Without Mutual Reactance", Proc. Symp. Modern Network Synthesis, Polytechnic Inst. of Brooklyn, N.Y., pp. 79-97; 1955.

3. Fialow, A. and I. Gerst, "The Transfer Function of General Two Terminal Pair RC Networks", Quart. Appl. Math., vol. 10, no. 2, pp. 113-127; July, 1952.

4. Fialkow, A. and I. Gerst, "The Transfer Function of Networks Without Mutual Reactance", Quart. Appl. Math., vol 12, no. 2, pp. 117-131; July, 1954.

5. Fialkow, A., D. Hazony, P. Kodali, and T. Zeren "Transfer-Function Realizability of Multiport RLC Transformerless Grounded Networks", IEEE Trans. on Circuit Theory, vol. CT-11, no. 1, pp. 73-79; March 1964.

6. Hazony, D., Elements of Network Synthesis, Reinhold Publ. Corp., New York, New York; 1963.

7. Hazony D. and R.D. Joseph, "Transfer Matrix Synthesis Using Active RC Networks", Case Institute of Technology, Scientific Report No. 9, Contract AF19(628)1699; May, 1965.

8. Kodali, P., "Variations on the Fialkow-Gerst Synthesis Technique", Master's Thesis, Case Institute of Technology, Scientific Report No. 1, Contract AF19(628)1699; December, 1962.

9. Zeren, T., "Multiport Voltage Transfer Function", Doctoral Thesis, Case Institute of Technology, Scientific Report No. 3, Contract AF19(628)1699; October, 1963 (Published in part as (5) above.).

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
CASE INSTITUTE OF TECHNOLOGY
Cleveland, Ohio

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP

3. REPORT TITLE
"Optimization of the Fialkow-Gerst Multiport RC Transfer Function Synthesis"

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
SCIENTIFIC REPORT NO. 10 (1965)

5. AUTHOR(S) *(Last name, first name, initial)*
Hazony, D. and Hilberman, D.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
1965	74	9

8a. CONTRACT OR GRANT NO. AF19(628)1699
b. PROJECT AND TASK NO. 5635, 01
c. DOD ELEMENT 61445014
d. DOD SUBELEMENT 681305

9a. ORIGINATOR'S REPORT NUMBER(S)
Scientific Report No. 10

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
AFCRL - 65-555

10. AVAILABILITY/LIMITATION NOTICES Qualified requestors may obtain copies of this report from DDC. Other persons or organizations should apply to the Clearinghouse for Federal Scientific and Technical Information (CFSTI), Sills Bldg. 5285 Port Royal Rd., Springfield, Virginia, 22151.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
Hq. AFCRL, OAR (CRD)
United States Air Force
L.G. Hanscom Field, Bedford, Mass.

13. ABSTRACT

"A method is presented which reduces the number of components needed in a Fialkow-Gerst multiport RC transfer function synthesis. A relationship is determined between the number of non-zero numerator coefficients in the transfer function vector and the number of components used in the synthesis of that vector".

DD FORM 1 JAN 64 1473

UNCLASSIFIED
Security Classification

UNCLASSIFIED
Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Fialkow-Gerst Synthesis						
Transfer Matrix Synthesis						
RC Networks						
Network Theory						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

UNCLASSIFIED
Security Classification